



ulm university

universität
uulm

Fakultät für Ingenieurwissenschaften und Informatik
Institut für Datenbanken und Informationssysteme

Master-Thesis
im Studiengang Informatik

Funktionsanalyse und Funktionserweiterung für eine personalisierte Mobile Crowd Sensing Plattform mit Algorithmen-Entwicklung

vorgelegt von

Johanna Ditzig

Januar 2016

1. Gutachter	Prof. Dr. Manfred Reichert
2. Gutachter	Dr. Winfried Schlee
Betreuer:	Dr. Rüdiger Pryss
Matrikelnummer	851365
Arbeit vorgelegt am:	07.01.2016

Kurzfassung

Mit dem Siegeszug des Smartphones im Jahr 2007 kommen mobile Applikationen, sogenannte Apps im Gesundheits-Bereich zunehmend zum Einsatz. In der Tinnitus-Forschung dienen solche Applikationen beispielsweise zur Datenerfassung von Tinnitus-Betroffenen. Die App TrackYourTinnitus kommt Betroffenen zur Dokumentation von Schwankungen der Tinnitus-Wahrnehmung zugute.

Innerhalb dieser Arbeit wird ein Konzept erarbeitet und realisiert, das die bestehende TrackYourTinnitus-Plattform um ein Formular für Administratoren zum Anlegen von Feedbacks und um einen Feedback-Algorithmus erweitert. Der Feedback-Algorithmus teilt Tinnitus-Betroffene entsprechend der Einflussfaktoren: Stress, Konzentration, Aufregung und Emotion verschiedenen Gruppen zu. Basierend auf dieser Gruppenzuweisung erhalten die Tinnitus-Betroffenen in der Android-App entsprechendes Feedback für den Umgang mit dem Tinnitus. Jede Woche erhalten die App-Anwender zusätzlich einen neuen Tipp der Woche. Alle Feedbacks können vom App-Anwender bewertet werden, sodass Rückschlüsse dahingehend gewonnen werden können, welches Feedback für welche Gruppe hilfreich ist.

Vorwort

Diese Arbeit markiert den Abschluss meines Studiums, das innerhalb der letzten Jahre durch viele spannende Projekte zur fachlichen und persönlichen Entwicklung beigetragen hat. Das Thema Funktionsanalyse und Funktionserweiterung für eine personalisierte Mobile Crowd Sensing Plattform mit Algorithmen-Entwicklung hat ein ideales Feld geboten, um medizinische und informationstechnische Themen zusammen zu bringen und persönlich fachübergreifend in Bezug zum Thema Tinnitus dazuzulernen. An dieser Stelle möchte ich mich bei allen Personen für die Unterstützung während der Anfertigung dieser Arbeit bedanken. Mein Dank gilt Prof. Dr. Manfred Reichert der Universität Ulm sowie dem Tinnituszentrum der Universität Regensburg für die Chance der Durchführung der Abschlussarbeit in einem praxisnahen und interessanten Themengebiet.

Besonderen Dank an Dr. Rüdiger Pryss (Institut für Datenbanken und Informationssysteme, Ulm) und Dr. Winfried Schlee (Tinnituszentrum Universität Regensburg), die zum Gelingen dieser Arbeit beigetragen haben. Für die Betreuung, konstruktiven Diskussionen, die hilfreichen Anregungen und die stets motivierende Rückmeldung möchte ich mich herzlich bedanken. Vielen Dank auch an Johannes Schobel (Institut für Datenbanken und Informationssysteme, Ulm) für die Projekteinführung und hilfreichen Hinweise zum Laravel-Framework sowie an Alexander Bachmeier (Institut für Datenbanken und Informationssysteme, Ulm) für die Bereitstellung projektrelevanter Informationen. Ein großes Dankeschön auch an alle, die am Gegenlesen dieser Arbeit beteiligt waren.

Darüber hinaus möchte ich mich herzlich bei meinen lieben Eltern, meiner Familie und meinen Freunden für die moralische Unterstützung und den Rückhalt über die Dauer des Studiums bedanken.

Eigenständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Hilfsmittel verwendet habe. Sinngemäße Übernahmen aus anderen Werken sind als solche kenntlich gemacht und mit genauer Quellenangabe (auch aus elektronischen Medien) versehen.

Ulm, 07.01.2016

Johanna Ditzig

Inhaltsverzeichnis

1. Einleitung	1
1.1. Motivation	1
1.2. Problemstellung	4
1.3. Zielsetzung	4
1.4. Aufbau der Arbeit	5
2. Grundlagen	7
2.1. Tinnitus	7
2.2. TrackYourTinnitus	8
2.2.1. Mobile Crowd Sensing	8
2.2.2. TrackYourTinnitus-Architektur	8
2.2.3. TrackYourTinnitus-Funktionalität	9
2.3. Stand der Forschung	10
3. Analyse und Anforderungsdefinition	13
3.1. Analyse bestehender Mobile-Health-Apps	13
3.1.1. Symptomate Symptom Checker	13
3.1.2. Mein Kopfschmerz	15
3.1.3. CatchMyPain	16
3.1.4. Metappbolic	18
3.2. Datenanalyse	18
3.3. Funktionsanalyse	20
3.3.1. Szenario	20
3.3.2. Anwendungsfalldiagramme	21
3.4. Anforderungsdefinition	22
3.4.1. Funktionale Anforderungen	23
3.4.2. Nichtfunktionale Anforderungen	25
4. Konzept	27
4.1. Umsetzungsoptionen	27
4.2. Hauptgruppendefinition	28
4.2.1. Korrelationskoeffizienten	28
4.2.2. Einflussfaktoren	30
4.3. Feedback-Arten	32
4.4. Feedback-Algorithmus	33
4.4.1. Hauptgruppen-Algorithmus	33
4.4.2. Regeldefinition und -auswertung	36
4.5. Usability-Konzept	37
4.5.1. Feedback-Formular	37
4.5.2. Android-App	40
5. Architektur und Implementierung	43
5.1. Gesamtübersicht	43

5.2. Datenstrukturen	44
5.3. Feedback-Formular	47
5.4. Android-Applikation	49
5.4.1. Einflussfaktoren	50
5.4.2. Tipp der Woche	55
5.4.3. Feedback bewerten	57
5.4.4. Feedback geben	58
5.5. Schnittstellen	58
5.5.1. Einflussfaktoren	58
5.5.2. Tipp der Woche	59
5.5.3. Feedback-Bewertung	59
5.6. Feedback-Algorithmus	60
5.6.1. Hauptgruppen-Algorithmus	61
5.6.2. Regelauswertung	69
6. Anforderungsabgleich	73
6.1. Funktionale Anforderungen	73
6.2. Nichtfunktionale Anforderungen	75
7. Zusammenfassung und Ausblick	77
7.1. Zusammenfassung	77
7.2. Ausblick	78
7.2.1. Erweiterung des Feedback-Algorithmus	78
7.2.2. Erweiterung der App-Funktionalität	79
7.2.3. Erweiterung der TrackYourTinnitus-Plattform	80
A. Anhang	81

Abbildungsverzeichnis

1.1. Internetnutzung - erstes Quartal 2014 [Des14b]	1
1.2. Marktanteile mobiler Betriebssysteme [Sta15]	2
1.3. Gesundheitsapps nach Therapiebereich im Jahr 2013 [Sta13]	3
2.1. TYT-Architektur nach [PRLS15]	9
3.1. Symptomate Symptom Checker - Datenerhebung [Inf14]	14
3.2. Symptomate Symptom Checker - Ergebnisse [Inf14]	14
3.3. Mein Kopfschmerz [Pfi15]	15
3.4. CatchMyPain-Schmerzkurve [San15]	16
3.5. CatchMyPain-Schmerzeintrag [San15]	17
3.6. CatchMyPain-Community [San15]	17
3.7. metappbolic [Aki14]	18
3.8. Datenanalyse - Ausschnitt der Datenstruktur	19
3.9. Szenario - Ablauf Feedback-Algorithmus	20
3.10. Anwendungsfalldiagramm - Administrator	21
3.11. Anwendungsfalldiagramm - Anwender	22
4.1. Tinnitus-Gruppen - Einflussfaktor Stress	31
4.2. p-Wert - Einflussfaktor Stress	31
4.3. Tinnitus-Gruppen - Einflussfaktor Konzentration	32
4.4. Mockup - Feedback-Liste	38
4.5. Mockup - Feedback-Formular	39
4.6. App-Mockups mit Navigationskonzept	40
5.1. Architektur-Gesamtübersicht	44
5.2. Datenstruktur - Feedbacks	45
5.3. Datenstruktur - Hauptgruppen	46
5.4. Feedback-Formular	47
5.5. Beispiel - Regeldefinition	48
5.6. Beispiel - Optionen zur Regeldefinition	49
5.7. Screenshot - Hauptmenü Android-App	50
5.8. Ablaufdiagramm - Einflussfaktoren	51
5.9. Screenshot - Einflussfaktoren und Feedback-Ansicht	52
5.10. Screenshot - Tipp der Woche	55
5.11. Ablaufdiagramm - Tipp der Woche	56
5.12. Ablaufdiagramm - Hauptgruppen-Algorithmus	61
5.13. Beispiel - Entfernung von NULL-Werten	64
5.14. Beispiel - Sortierung der Datenwerte der Tinnitus-Lautheit	65
5.15. Beispiel - DOR-Berechnung durch Aufteilung der Datenwerte der Tinnitus-Lautheit	66
5.16. Ablaufdiagramm - Feedback-Ermittlung	70
5.17. Beispiel - Regelauswertung	72

Tabellenverzeichnis

3.1. Funktionale Anforderungen	23
3.2. Nichtfunktionale Anforderungen	25
4.1. Hauptgruppendefinitionen	30
4.2. Feedback-Arten	33
4.3. DOR-Wertebereich	35
6.1. Erfüllungsgrad funktionale Anforderungen	74
6.2. Erfüllungsgrad nichtfunktionale Anforderungen	75
A.1. App-Fragebogen [Sch14]	81
A.2. Mini Tinnitus Fragebogen (Mini-TF) [HG04]	81
A.3. Tinnitus Sample Case History Questionnaire (TSCHQ) [Lan06]	82
A.4. Schlimmstes Symptom [Sch13]	84

Listings

5.1. Datenbankaufruf von Hauptgruppen	48
5.2. Hauptgruppenanzeige in Feedback-Formular	48
5.3. Feedback in Datenbank speichern	49
5.4. Feedback-Download	52
5.5. Feedback-Anzeige	53
5.6. Klick-Listener für Listenelemente	54
5.7. Anzeige Feedback-Details	55
5.8. Tipp der Woche anzeigen	57
5.9. Feedback-Bewertung senden	57
5.10. Ausgabeformat der Schnittstelle "Einflussfaktoren"	58
5.11. Ausgabeformat der Schnittstelle "Tipp der Woche"	59
5.12. Eingabeformat der Schnittstelle "Feedback-Bewertung"	60
5.13. Feedbackbewertung speichern	60
5.14. Hauptgruppen-Algorithmus - Ermittlung des Datenzeitraums	62
5.15. Hauptgruppen-Algorithmus - Werte berechnen	67
5.16. Hauptgruppen-Algorithmus - Feedback vergeben	69
5.17. Regelauswertung	71

1

Einleitung

1.1. Motivation

Die Welt in 100 Jahren: „Es wird jedermann sein eigenes Taschentelephon haben, durch welches er sich, mit wem er will, wird verbinden können, einerlei, wo er auch ist ... Die Bürger jener Zeit werden überall mit ihrem drahtlosen Empfänger herumgehen ... Monarchen, Kanzler, Diplomaten, Bankiers, Beamte und Direktoren werden ihre Geschäfte erledigen können, wo immer sie sind ... Alle diese Wunder der drahtlosen Telegraphie werden das kommende Zeitalter zu einem grossartigen, unglaublichen machen.“ [Mat05]

Diese Utopie des „drahtlosen Jahrhunderts“ aus dem Jahr 1910, in der sich Funk- und Telefontechnik gerade erst in den Anfängen befunden haben, ist längst Realität. Mobiltelefone und Internet sind durch technologische Entwicklungen in der Mikroelektronik, Informationstechnologie und Kommunikationstechnik gegenwärtig und in Industrie, Wirtschaft und Gesellschaft nicht mehr weg zu denken [Mat05]. Die Relevanz des Internets zeigt sich durch dessen Nutzung. Etwa 80% der Menschen (58,6 Millionen) haben das Internet im ersten Quartal 2014 verwendet [Des14b]. Im Jahr 2010 lag der Nutzungsanteil vergleichsweise bei 75% [Des14b]. Abbildung 1.1 veranschaulicht die Internetnutzung eingeteilt nach Altersgruppen.

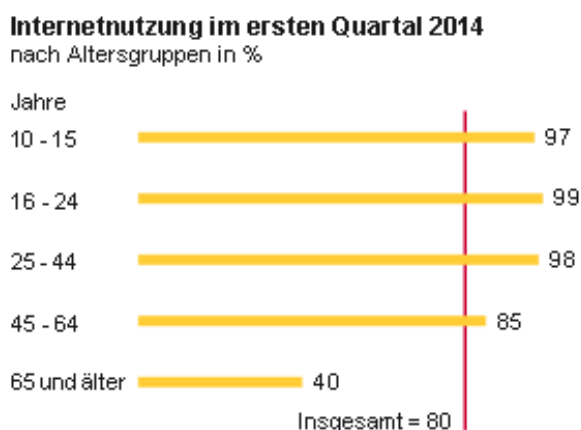


Abbildung 1.1.: Internetnutzung - erstes Quartal 2014 [Des14b]

Neben der Internetnutzung gewinnt die mobile Internetnutzung zunehmend an Bedeutung. Dabei werden Handys, Smartphones sowie Laptops oder Tablet-Computer über drahtlose Netzwerke oder per Mobilfunknetz verwendet. Im ersten Quartal 2014 haben 63% der

Internetnutzer das mobile Internet benutzt [Des14a]. Aufgrund der standortunabhängigen mobilen Such- und Informationsmöglichkeiten sind Smartphones zu einem wichtigen Bestandteil des Alltags geworden. Ein Großteil der Befragten verwendet das mobile Internet mehrmals täglich und immer mehr bevorzugen es im Vergleich zum stationären Internet bedingt durch Gründe, wie Bequemlichkeit, leichter Zugriff, Zeitersparnis und die Nutzung von Applikationen, sogenannter Apps [TOM15]. Das mobile Internet wird am häufigsten eingesetzt um Webseiten und Apps aufzurufen [TOM14]. Hierbei spielen Bedienbarkeit und Nutzbarkeit eine wesentliche Rolle. Zu den häufigsten verwendeten Apps zählen Dienstprogramme und Nachrichten-Apps [TOM14]. Durchschnittlich haben die Befragten 11 bis 22 Apps auf ihrem Gerät installiert [TOM15].

Das heutige Zeitalter der sogenannten „Post-PC-Ära“¹ wirft bei der Entwicklung mobiler Applikationen Herausforderungen dahingehend auf, dass die Heterogenität mobiler Plattformen berücksichtigt werden muss. Sollen Applikationen auf einer Vielzahl an Plattformen vertreten sein, ist die Entwicklung besonders zeit- und kostenintensiv. Daher ist eine im Vorfeld stattfindende Auseinandersetzung dahingehend erforderlich, für welche Betriebssysteme mobile Applikationen unterstützt werden sollen, um viele Endanwender zu erreichen [WIM12]. Derzeit verfügt das mobile Betriebssystem Android über den größten Marktanteil, gefolgt von iOS. Abbildung 1.2 veranschaulicht die Marktanteile von Smartphone-Betriebssystemen basierend auf den Verkäufen von Januar 2012 bis März 2015 in Deutschland [Sta15].

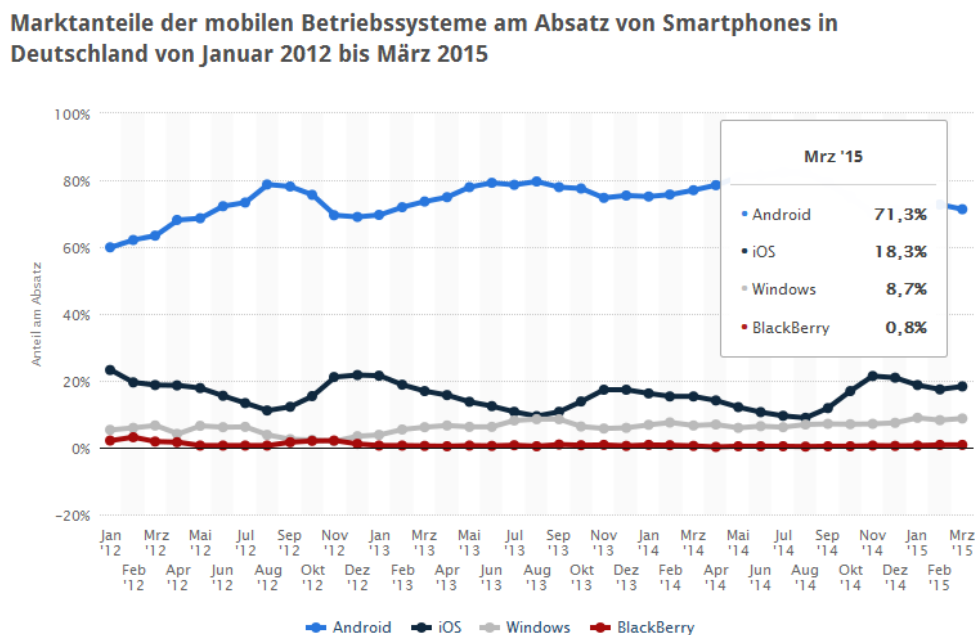


Abbildung 1.2.: Marktanteile mobiler Betriebssysteme [Sta15]

In den zugehörigen App-Stores existieren eine Vielzahl an Apps aus verschiedenen Bereichen, wie z.B. Musik, soziale Netzwerke, Spiele, Navigation, Bildung, Bücher sowie Gesundheit und

¹Unter Post-PC-Ära ist der Paradigmenwechsel vom Desktop-Internet zu Smartphone-Applikationen und Tablet-Systemen zu verstehen [WIM12].

Fitness. Bis 2017 werden schätzungsweise weltweit 3,4 Milliarden Menschen ein Smartphone besitzen und jeder zweite davon verwendet Mobile-Health-Apps [Eur15]. Beispiele für Mobile-Health (mHealth) Dienste sind Apps, die Ernährungs- oder Fitnessempfehlungen zur Verbesserung des Gesundheitszustands geben, die Patienten an das Einnehmen von Arzneimitteln erinnern oder die Vitalparameter, wie z.B. den Blutdruck messen. Abbildung 1.3 zeigt die verschiedenen Bereiche von Gesundheits-Apps [Sta13]. Einen großen Teilbereich nehmen hierbei Apps für Augen und Ohren ein. Im Bereich der mHealth-Apps haben Apps, die der Therapiesteuerung bzw. der Kontrolle des Verlaufs von Patienten dienen das größte Marktpotential [Res14].

Große Vielfalt im Bereich der Gesundheitsapps

Anzahl der Gesundheitsapps nach Therapiebereich im Jahr 2013

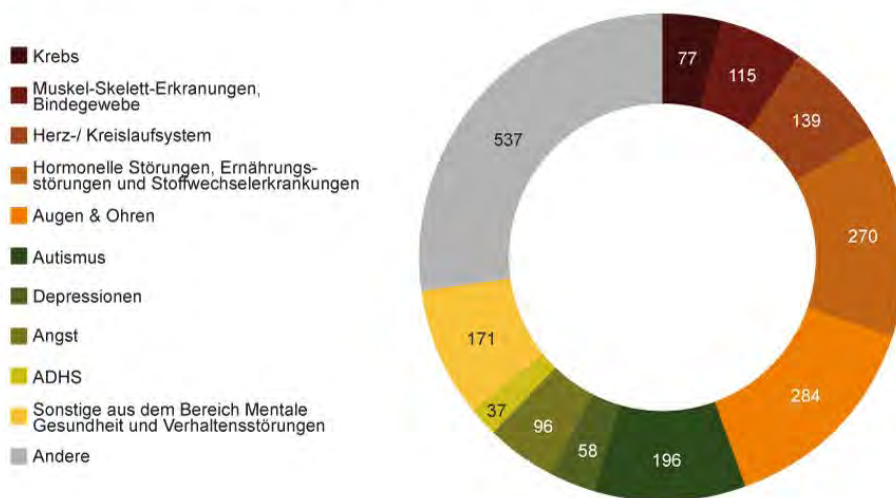


Abbildung 1.3.: Gesundheitsapps nach Therapiebereich im Jahr 2013 [Sta13]

Weiter eignen sich mHealth-Apps für die Forschung im Gesundheitswesen. Beispielsweise können Kliniken mHealth-Apps für die Aufnahme und Behandlung von Patienten verwenden [PMLR15]. Zudem bieten solche Apps einen Weg zur Erfassung von Patientendaten für Forschungszwecke [RLPL⁺13][IRLP⁺13].

Etwa drei Millionen Menschen sind in Deutschland von Tinnitus betroffen. Als gebräuchliches Synonym wird der Begriff "Ohrensausen" verwendet. Bei Tinnitus werden Geräusche wahrgenommen, ohne das Einwirken eines akustischen Reizes [Tin15]. Steigende Ansprüche an mobile Apps und eine zunehmende Zahl an Lösungen im Mobile-Health-Bereich, wie beispielsweise die App Tinnitracks [Son15], die Musik für Tinnitus-Betroffene entsprechend ihrer Tinnitus-Frequenz optimiert, setzen eine Auseinandersetzung mit den Möglichkeiten zur optimalen Unterstützung von Tinnitus-Betroffenen sowie der Tinnitus-Forschung durch Smartphones voraus und stellen einen zentralen Punkt dieser Arbeit dar [SRP⁺15].

1.2. Problemstellung

Die Anzahl an Personen, die in Deutschland von Tinnitus betroffen sind, zeigt die Relevanz der Tinnitus-Forschung. Betroffene leiden über Jahre hinweg an Tinnitus. Die Lebensqualität der Betroffenen ist durch das Symptom stark beeinträchtigt. Das Leiden kann unter anderem zu sozialem Rückzug, Depressionen oder zum Suizid führen. Selbst bei einer Vielzahl an vorhandenen Therapiemöglichkeiten, die entweder direkt auf den Tinnitus einwirken sollen, indem sie die Tinnitus-Lautheit reduzieren oder Therapien, die die Einstellung der Betroffenen zu ihrem Tinnitus positiv beeinflussen sollen, kann derzeit keine allgemeingültige Therapieform herangezogen werden, die allen Tinnitus-Betroffenen hilft. Mögliche Therapien sind z.B. medikamentöse Behandlungen, die Unterstützung des Tinnitus-Betroffenen bei der Bewältigung der Tinnitus-Belastung (Coping), die Reduzierung der Angst zu dem Symptom Tinnitus durch adäquate Krankheitsinformationen (Counseling), die Gewöhnung des Betroffenen an die Ohrgeräusche (Habituation) oder das Verlernen der Tinnitus-Wahrnehmung (Retraining) [Bie07].

Tinnitus-Betroffene benötigen Informationen und Ratschläge, wie sie damit umgehen können, denn Tinnitus kann sich auf verschiedene Arten auswirken und nicht bei jedem Betroffenen führt die selbe Behandlung zu Erfolg. Für einen Betroffenen müssen über eine längere Zeit Daten erfasst werden, die Rückschlüsse zu den Einflussfaktoren des Tinnitus geben.

Die vorliegende Arbeit soll die Frage beantworten, wie für Tinnitus-Betroffene durch den Einsatz einer Mobile Crowd Sensing Plattform (vgl. Kapitel 2.2.1) und die dadurch erfassten Daten Rückmeldung (engl. Feedback) gegeben werden kann. Hierzu müssen die Einflussfaktoren des Tinnitus ermittelt werden und daraus Handlungsvorschläge gegeben werden, die im Umgang mit dem Tinnitus helfen können. Zentraler Punkt der Arbeit ist die Forschungsfrage: *"Wie muss ein Feedback-Algorithmus für Tinnitus-Betroffene implementiert sein, um über eine mobile Applikation Feedbacks zu geben?"*. Daraus lassen sich Fragen dahingehend ableiten, welche Daten für ein Feedback in Frage kommen, wie Feedback gegeben werden kann und welche Feedback-Arten.

1.3. Zielsetzung

Ausgangspunkt dieser Arbeit ist eine Mobile Crowd Sensing Plattform, deren Konzeption, Design und Implementierung innerhalb mehrerer studentischer Projekte der Universität Ulm in Kooperation mit dem Tinnituszentrum Regensburg erfolgte [Tin15][PRLS15]. Die Plattform besteht aus einer Webseite sowie aus nativen mobilen Anwendungen für die Plattformen Android und iOS. Tinnitus-Betroffene können sich sowohl über die Webseite, als auch per App registrieren. Für Forschungszwecke und um Rückschlüsse auf die Schwankungen der Tinnitus-Wahrnehmung des Betroffenen zu erzielen, bekommt dieser in definierten Abständen per App Fragebögen zum Ausfüllen. Der bisherige Stand bietet Betroffenen allerdings keine Rückmeldung dahingehend, wie sie handeln können, um ihren Zustand zu verbessern.

Das Ziel der Arbeit besteht in einer „Funktionsanalyse und Funktionserweiterung für eine personalisierte Mobile Crowd Sensing Plattform mit Algorithmen-Entwicklung“ im Kontext der Tinnitus-Forschung. Hierbei sollen zum einen bereits vorhandene Daten, die Patienten über Fragebögen ausgefüllt haben analysiert werden und anschließend ein Feedback-Algorithmus

implementiert werden. Dies soll den App-Anwendern einen Mehrwert bieten, indem sie darüber informiert werden, was die gegebenen Antworten zu den Fragebögen bedeuten, indem eine Auswertung der Daten stattfindet. Zudem soll ein App-Anwender Feedbacks erhalten, die ihn dabei unterstützen seine individuellen Tinnitus-Beschwerden zu lindern. Weiter grenzt sich diese Arbeit dahingehend ab, dass die Feedback-Anzeige lediglich für die Android-App realisiert wird.

1.4. Aufbau der Arbeit

Die vorliegende Arbeit besteht aus sieben Kapiteln.

Kapitel 2 bezieht sich auf die zur Realisierung erforderlichen Grundlagen. Es gibt Aufschluss zum Themengebiet Tinnitus. Zudem erläutert es die Plattform TrackYourTinnitus (TYT) sowie was unter einer Mobile Crowd Sensing Plattform zu verstehen ist. Weiter beschreibt es den aktuellen Stand der Forschung im Bereich der Gruppeneinteilung von Tinnitus-Patienten.

Kapitel 3 stellt die Vorgehensweise vor, durch die ein Konzept für einen Feedback-Algorithmus entwickelt werden soll. Dabei analysiert es bestehende mHealth-Apps dahingehend, in welchem Ausmaß diese den Anwendern Rückmeldung geben und welche Funktionalitäten sich für das bestehende TrackYourTinnitus-System eignen. Ferner stellt es den bestehenden Daten-Ist-Zustand vor und es zeigt die funktionalen und nichtfunktionalen Anforderungen an die zu entwickelnden System-Erweiterungen.

Kapitel 4 stellt das Konzept vor, das der Implementierung des Feedback-Algorithmus dient. Dazu stellt es verschiedene Umsetzungsoptionen vor und beschreibt, wie Tinnitus-Betroffene zu Gruppen klassifiziert werden können und wie anhand der bestehenden Daten Feedbacks gegeben werden können. Weiter beschreibt es verschiedene Feedback-Arten, sowie das Konzept für den zu realisierenden Feedback-Algorithmus. Zudem stellt es ein Usability-Konzept für das zu entwickelnde Feedback-Formular zum Anlegen von Feedbacks und die Android-App vor.

Kapitel 5 stellt eine Gesamtübersicht der zu erweiternden Architektur vor und beschreibt, wie die Datenstrukturen erweitert werden. Weiter beschreibt es die Implementierung des Feedback-Formulars, der Android-Applikation, der Schnittstellen, sowie des Feedback-Algorithmus, der aus Hauptgruppen-Algorithmus und einer Regelauswertung besteht.

Kapitel 6 stellt einen Anforderungsabgleich auf und beschreibt den Erfüllungsgrad der definierten funktionalen und nichtfunktionalen Anforderungen aus Kapitel 3.

Kapitel 7 fasst die Ergebnisse der Arbeit zusammen und gibt einen Ausblick auf zukünftige Erweiterungspotentiale.

2

Grundlagen

Dieses Kapitel beschreibt projektrelevante Grundlagen. Kapitel 2.1 gibt eine Übersicht in Bezug auf Tinnitus, indem das Symptom beschrieben wird. Kapitel 2.2 beschreibt die Mobile Crowd Sensing Plattform TrackYourTinnitus, die für diese Arbeit die Ausgangsbasis darstellt. Kapitel 2.3 legt den Stand der Forschung in Hinblick auf bestehende Ansätze dar, die Tinnitus-Betroffene in Gruppen einteilen.

2.1. Tinnitus

Der Fachbegriff Tinnitus (aus dem Lateinischen: tinnire = tönen, brummen, klingeln) bezeichnet alle Hör-Wahrnehmungen, die Betroffene ohne externen Reiz wahrnehmen [Bie05][Bie07]. Etwa 5-10% der Gesamtbevölkerung leiden unter Tinnitus und geben an, dass sie diesen fortlaufend wahrnehmen [SHP⁺14a]. Unterschieden wird zwischen einem objektiven und subjektiven Tinnitus. Ein objektiver Tinnitus ist seltener und kann von Geräuschen innerhalb des Körpers stammen [KVL13]. Ein subjektiver Tinnitus beschreibt die Wahrnehmung von Phantomgeräuschen [KVL13]. Er kann unterschiedlich verteilt wahrgenommen werden, entweder auf einem Ohr, beiden Ohren oder im Inneren des Kopfes oder an anderer Stelle. Auch unterscheidet sich der wahrgenommene Ton und tritt beispielsweise als Pfeifen oder Rauschen auf. Bei einem subjektiven Tinnitus kann weiterhin zwischen einem akuten Tinnitus (dauert weniger als 3 Monate), einem subakuten Tinnitus (dauert zwischen 3 bis zu 6 Monaten) und einem chronischen Tinnitus (dauert länger als 6 Monate) unterschieden werden [Bie07]. Betroffene die unter einem chronischen Tinnitus leiden, lassen sich in Betroffene einteilen, die mit ihrem Tinnitus zurechtkommen, da sie sich beispielsweise an ihn gewöhnen (kompensierter Tinnitus) sowie in Betroffene, die verstärkt unter dem Tinnitus leiden und dauerhaft beeinträchtigt sind (dekompensierter Tinnitus) [Bie05]. Tinnitus kann zu ernstzunehmenden psychologischen Folgen, wie Depressionen, Schlaflosigkeit und Angstzuständen, führen [SHP⁺14b].

Tinnitus-Beschwerden können in folgende vier Schweregrade eingeteilt werden [Bie05]:

- Grad 1: Der Tinnitus ist gut kompensiert, es besteht kein Leidensdruck.
- Grad 2: Der Tinnitus wirkt zeitweise störend in bestimmten Situationen und bei bestimmten Belastungen.
- Grad 3: der Tinnitus führt zu einer dauernden Beeinträchtigung im privaten und/oder beruflichen Bereich. Es treten Störungen im emotionalen und körperlichem Bereich auf, aber es besteht noch Arbeitsfähigkeit.

- Grad 4: Der Tinnitus führt zu einer völligen Dekompensation im privaten Bereich und zur Erwerbsunfähigkeit.

Bei der Behandlung von chronischem, subjektivem Tinnitus sind Therapiegespräche und die Aufnahme der vorhandenen Symptome und Ursachen, die zu dem Tinnitus führen sehr wichtig [KVL13].

2.2. TrackYourTinnitus

Die Plattform *TrackYourTinnitus*¹ (TYT) ist zur Ermittlung von Schwankungen der Tinnitus-Wahrnehmung durch die Universitäten Ulm und durch das Tinnituszentrum Regensburg entwickelt worden [PRLS15]. Bei TYT handelt es sich um eine *Mobile Crowd Sensing Plattform* (vgl. Kapitel 2.2.1), die durch den Einsatz spezieller Fragebögen die individuelle Wahrnehmung des Tinnitus im alltäglichen Leben der Tinnitus-Betroffenen verfolgt und aufzeichnet. Außerdem können die Umgebungsgeräusche aufgenommen werden, während der Tinnitus-Betroffene die Fragebögen ausfüllt. Alle Daten werden anschließend auf einem Server gespeichert, um später verarbeitet und analysiert werden zu können. Ziel von TYT ist es Schwankungen der Tinnitus-Wahrnehmung und der Tinnitus-Auswirkungen von Betroffenen zu messen, um daraus Rückschlüsse auf die Ursachen des Tinnitus, in Verbindung mit dem alltäglichen Leben der Tinnitus-Betroffenen ziehen zu können [PRLS15][PRH⁺15]. Weiter lässt sich zur Behandlung von Tinnitus erforschen, wie Tinnitus-Betroffene anhand ihrer Symptome gruppiert werden können.

2.2.1. Mobile Crowd Sensing

Unter *Mobile Crowd Sensing* (MCS) versteht man das Paradigma große Datenmengen über weit verbreitete mobile Geräte zu sammeln. MCS wird durch die Verwendung von Smartphones, die mit einer Vielzahl von Sensoren ausgestattet sind, ermöglicht [LML⁺10]. Durch den Einsatz mobiler Smartphones können Plattformen wie TYT viele Menschen erreichen und somit Daten erfasst werden, die andernfalls schwer zu ermitteln sind. Die Strukturierung der Daten spielt bei der Datensammlung durch mobile Geräte eine wichtige Rolle [SPR15], da die Daten später verarbeitet werden müssen. Im Hintergrund einer MCS-Plattform wird ein entsprechendes Backend benötigt, das die gesammelten Daten speichert und verwaltet. Um die Effizienz bei der Erfassung der Fragebögen zu steigern, kann für die MCS-Plattform ein prozess-gestützter Ansatz zur Datensammlung durch Prozessmanagement Technologien verwendet werden [SSP⁺14].

2.2.2. TrackYourTinnitus-Architektur

Die TYT-Plattform besteht aus mobilen Applikationen für Android und iOS, einer Webseite [Sch13] und einem Backend [PRLS15]. Abbildung 2.1 stellt diese Komponenten dar und zeigt, wie sie zueinander in Verbindung stehen. Für die beiden Betriebssysteme Android und iOS sind Apps entwickelt, über die Daten der Tinnitus-Betroffenen erfasst werden können. Die Apps sind jeweils in den App-Stores erhältlich. Die Webseite ist mit Hilfe der Frameworks

¹<https://www.trackyourtinnitus.org>

Twitter Bootstrap und *Laravel* [Otw15] entwickelt und kommuniziert über *Sockets* mit dem Backend. Das Backend enthält eine *MySQL*-Datenbank und speichert darin sowohl die Anwenderdaten als auch die durch die Fragebögen erfassten Daten. Als Programmier-Modell wird hierbei das *Model-View-Controller*-Pattern² (MVC) eingesetzt, wobei die *View* durch die mobilen Applikationen und die Webseite repräsentiert wird und der *Controller* durch das *Laravel-Framework* realisiert ist. Die Programmiersprache, die beim Backend eingesetzt wird, ist *PHP* [PHP15]. Das Backend bietet außerdem eine REST-Schnittstelle an, über die die mobilen Applikationen Daten abfragen und senden können. Beispielsweise speichern die Apps die ausgefüllten Fragebögen über diese Schnittstelle. REpresentational State Transfer (REST) beschreibt weder Nachrichten noch Protokolle, sondern architektonische Elemente, die als Regeln angesehen werden können [Mel08]. Hierzu zählen Ressourcen, einheitliche Schnittstellen und Hypermedia. REST verwendet HTML als Übertragungsprotokoll und nutzt daher die vier Methoden: *GET* zum Abfragen von Ressourcen, *POST* zum Anlegen von Ressourcen, *PUT* zum Ändern von Ressourcen und *DELETE* zum Löschen von Ressourcen.

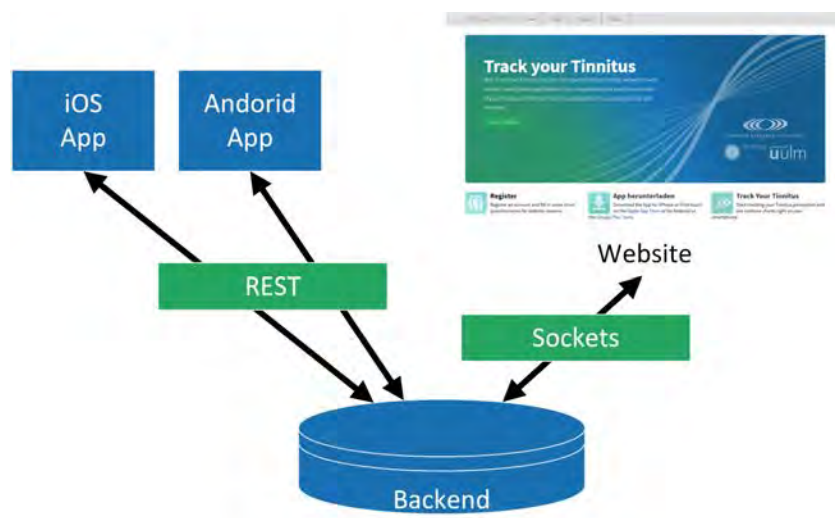


Abbildung 2.1.: TYT-Architektur nach [PRLS15]

2.2.3. TrackYourTinnitus-Funktionalität

Um TYT verwenden zu können, müssen sich Tinnitus-Betroffene zunächst an der Plattform registrieren. Zur Charakterisierung des Tinnitus müssen drei statistische Fragebögen von den Tinnitus-Betroffenen ausgefüllt werden (vgl. Kapitel 3.2). Zur Aufnahme der Tinnitus-Wahrnehmung und anderer relevanter Parameter, die sich auf den Tinnitus beziehen können, dient das Ausfüllen der App-Fragebögen. Diese Fragebögen sollen regelmäßig ausgefüllt werden, wobei der genaue Zeitpunkt, wann ein Fragebogen ausgefüllt wird unterschiedlich ist. Die Häufigkeit, wie oft und an welchen Wochentagen ein App-Fragebogen ausgefüllt werden soll, kann ein Tinnitus-Betroffener selbst in den Einstellungen definieren. Dadurch wird die Aufnahme der Daten in den Alltag integriert. Das Ausfüllen der App-Fragebögen ist lediglich

²Trennt die Programmierung eines Systems in die Bereiche Daten (Model), Anzeige (View) und Logik (Controller), um das System zu strukturieren.

über die mobilen Applikationen möglich. Die Ergebnisse der ausgefüllten Fragebögen werden in der App graphisch dargestellt und bieten jedem Tinnitus-Betroffenen eine individuelle Übersicht zu den Schwankungen seiner Tinnitus-Wahrnehmung.

2.3. Stand der Forschung

Um Patienten eine bessere Tinnitus-Behandlung zu vermitteln, machen [TCTJ08] den Vorschlag die Patienten in verschiedene Subgruppen zu unterteilen. Dieses Vorgehen resultiert aus der Beobachtung, dass verschiedene Patienten ihren Tinnitus unterschiedlich wahrnehmen. Beispielsweise werden die Symptome des Tinnitus bei manchen Patienten stärker und bei manchen schwächer, wenn die Umgebungsgeräusche zunehmen. Zur Festlegung der Subgruppen verwenden [TCTJ08] eine 2-Schritt Clusteranalyse, die auf 26 kategorischen und 25 fortführenden Variablen beruht und anhand von 256 Teilnehmern durchgeführt worden ist. Die Variablen sind dabei aus verschiedenen Fragebögen entnommen, die durch die Teilnehmer im Zuge klinischer Tests ausgefüllt worden sind. Durch diese Clusteranalyse identifiziert [TCTJ08] vier Cluster. Diese analysieren sie weiter anhand der *Tinnitus-Lautheit*, einer emotionalen Skala, einer Schlaf-Skala und einer Konzentrations-Skala, um die Cluster einzuordnen. Aus den Ergebnissen dieser Analysen geben [TCTJ08] folgende Subgruppen für Tinnitus-Patienten an:

- Konstanter, quälender Tinnitus.
- Schwankender Tinnitus, der bei Lärm schlimmer wird.
- Bewältigbarer Tinnitus, der durch Berührung nicht stärker wird.
- Bewältigbarer Tinnitus, der bei Stille stärker wird.

Die vier ermittelten Subgruppen sind dabei nicht als finale Gruppen festgelegt. [TCTJ08] stellen fest, dass ihre Arbeit die Relevanz der Bildung von Gruppen in der Tinnitus-Behandlung zeigt. Beim Vergleich von [TCTJ08] mit dem Vorgehen dieser Arbeit, bestätigt sich diese Aussage. Allerdings wird innerhalb dieser Arbeit ein anderes Verfahren verwendet, um Tinnitus-Betroffene in Gruppen zu unterteilen (vgl. Kapitel 4.2).

[ZRJT10] untersucht verschiedene Kombinationen wichtiger Faktoren, die zu einer signifikanten Verbesserung des Tinnitus bei Patienten geführt haben und setzt diese in Beziehung zu Tinnitus-Kategorien. Das Ziel dabei ist die Erhebung von Regeln, die Patienten bei der Milderung ihres Tinnitus helfen sollen. Die Datenbasis auf die sie sich dabei beziehen geht aus dem Verfahren der *Tinnitus Retraining Therapy*³ (TRT) hervor. TRT zielt darauf ab Patienten an die Tinnitus-hervorrufenden Reaktionen und Tinnitus-Wahrnehmung zu gewöhnen. Dabei verfolgt TRT das Ziel die Einwirkung des Tinnitus zu verringern oder sogar zu eliminieren [ZRJT10]. Die Kategorisierung der Patienten in verschiedene Tinnitus-Gruppen basiert auf Fragebögen, die sich auf Aktivitäten wie Schlaf, Arbeit und Konzentration beziehen und auf Tests, die ermitteln wie sich der Tinnitus des Patienten auf verschiedene Geräuschlevel auswirkt. Das Verfahren, das [ZRJT10] anwendet besteht aus fünf Schritten: Die Datenbasis wird zunächst gefiltert, um ungültige Datensätze auszuschließen. Daraufhin werden aus den gefilterten Daten Merkmale extrahiert und eine *K-Means* Clusteranalyse durchgeführt. Aus diesen Resultaten werden Entscheidungsbaum-Klassifikatoren erzeugt, die zur Regelgewinnung

³Zu deutsch: Tinnitus Umschulungs-Therapie

verwendet werden können. Nach diesem Prinzip führt [ZRJT10] Versuche durch, die als Resultat eine Reihe an Regeln erbringen. Diese Regeln bauen auf den Untersuchungen der Patienten im Rahmen von TRT auf. Das Ziel von [ZRJT10] ist im Gegensatz zu dieser Arbeit und [TCTJ08] weniger den Tinnitus in Gruppen zu kategorisieren, sondern Regeln zur Linderung von Tinnitus aufzustellen. Diese werden anhand von Faktoren, die bei anderen Patienten bereits zu einer Verbesserung des Tinnitus geführt haben, abgeleitet.

Analyse und Anforderungsdefinition

Zur Entwicklung des Konzeptes eines Feedback-Algorithmus analysiert Kapitel 3.1 bestehende Mobile-Health-Apps dahingehend, ob App-Anwender Feedback basierend auf getätigten Eingaben erhalten, bzw. wie Feedback-Interaktionen in solchen Apps realisiert sind. Zudem betrachtet Kapitel 3.2 den Ist-Zustand der vorhandenen Daten der TYT-Plattform, um Umsetzungspotentiale für den Feedback-Algorithmus treffen zu können. Durch ein Szenario und durch Anwendungsfälle erfolgt in Kapitel 3.3 eine Funktionsanalyse. Die daraus resultierenden Anforderungen werden in Kapitel 3.4 in der Anforderungsdefinition als funktionale und nichtfunktionale Anforderungen definiert.

3.1. Analyse bestehender Mobile-Health-Apps

Im Folgenden werden mHealth-Apps auf Funktionalität, Navigationsführung und Optik untersucht. Ziel ist es Anregungen für die zu entwickelnde Feedback-Funktion der TYT-Applikation zu finden und für App-Anwender eine intuitive Benutzerführung zu gestalten.

3.1.1. Symptomate Symptom Checker

Symptomate Symptom Checker [Inf14] ist eine Mobile-Health-App, die mit Ärzten entwickelt worden ist und Anwender dabei unterstützen soll mehr über bestimmte Symptome zu erfahren. *Symptomate Symptom Checker* bietet ausgewählte, aufeinanderfolgende Fragen zu individuellen Symptomen und basiert auf einem Algorithmus der künstlichen Intelligenz, bei dem eine medizinische Datenbank verwendet wird, die über 400 Bedingungen und über 1000 Symptome enthält. Zu den Diagnosen der App zählen unter anderen Migräne, Diabetes, Grippe und Spannungskopfschmerzen [Inf14].

Die App ermittelt Basisdaten zum Patienten, wie Geschlecht, Alter, Körpergröße und -gewicht. Daraus ermittelt sie zunächst durch listen-basierte Auswahloptionen allgemeine Symptome, wie Kopfschmerzen, Bauchschmerzen, Rückenschmerzen, Fieber, chronische Müdigkeit, Depression und Schwindel sowie weitere. Symptome können auch durch den Anwender selbst eingetragen werden. Durch aufeinander aufbauende Fragen überprüft der Algorithmus Risikofaktoren, die auf den Anwender zutreffen. Ferner werden die Regionen festgehalten, in denen der Anwender in den letzten 12 Monaten gelebt hat bzw. die er bereist hat. Abbildung 3.1 veranschaulicht Ausschnitte der Datenerhebung. Basierend auf den persönlichen Angaben erhält der Anwender, wie in Abbildung 3.2 dargestellt, Feedback. Hierzu zählen die Symptome, die auf ihn zutreffen, ein Richtwert, zu wie viel Prozent das Symptom zutrifft und ob das Aufsuchen eines Arztes dringend empfohlen ist. Weiter gibt die App Rückmeldung, auf Basis welcher Symptome das

Ergebnis bestimmt wird. Über die App kann der Anwender die Ergebnisse anfordern. Diese werden ihm per E-Mail als Bericht gesendet und können dem behandelnden Arzt vorgelegt werden. Um den Service zu verbessern kann der Anwender selbst die Ergebnisse beurteilen [Inf14].

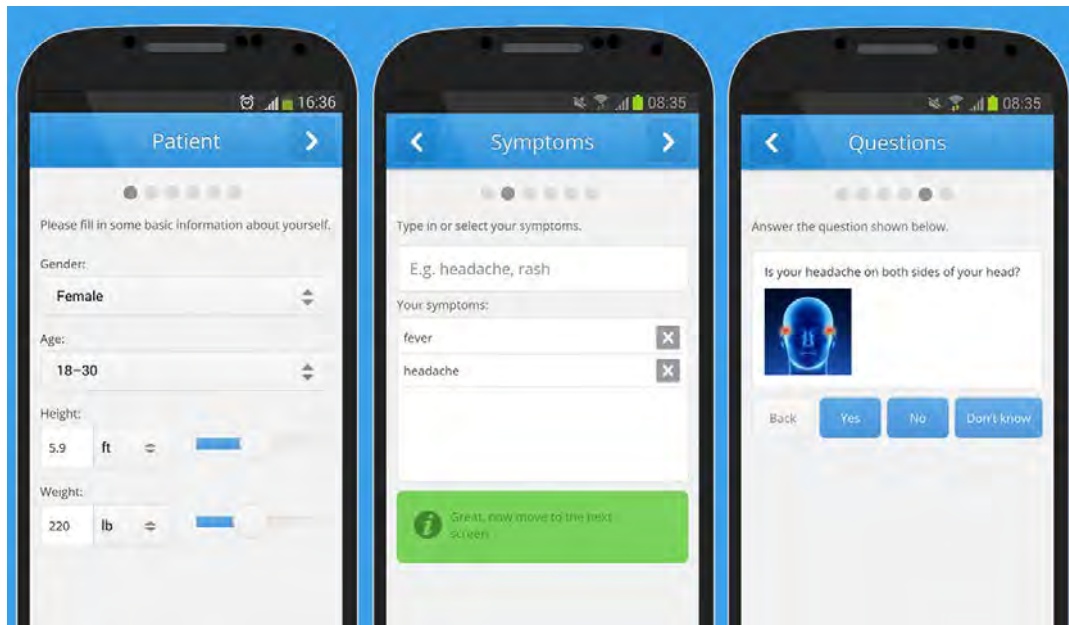


Abbildung 3.1.: Symptomate Symptom Checker - Datenerhebung [Inf14]

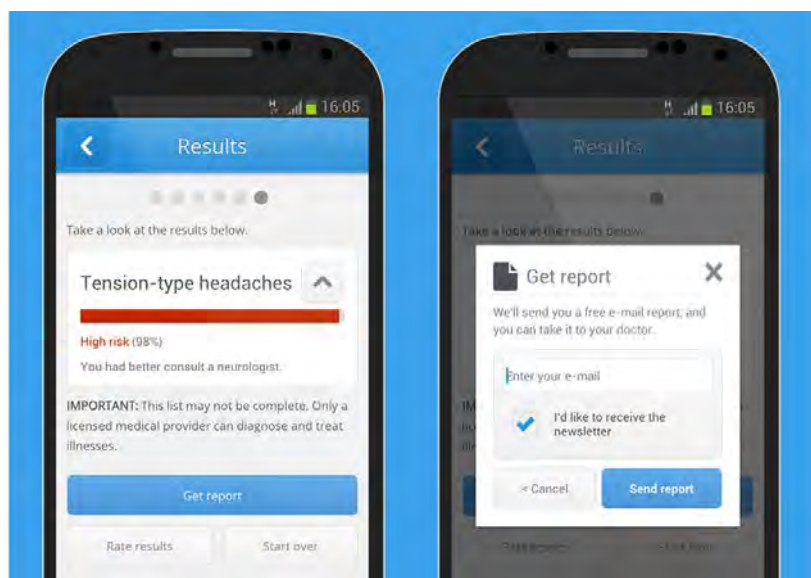


Abbildung 3.2.: Symptomate Symptom Checker - Ergebnisse [Inf14]

Symptomate Symptom Checker stellt ein gutes Beispiel für eine App mit Feedback-Funktion für die Anwender dar. Für das TrackYourTinnitus-Projekt sind folgende Optionen denkbar:

- Sofern die bestehenden Fragebögen erweitert bzw. neue Fragebögen hinzugefügt werden, lassen sich aufeinanderfolgende Abfragen relevanter Daten realisieren. Dadurch können auf Grund der ausgewählten Antworten des Tinnitus-Betroffenen Abfragezweige ausgeschlossen werden und unnötige Fragen ausgelassen werden.
- Der Tinnitus-Betroffene kann erhaltenes Feedback bewerten. Dies kann entweder dadurch erfolgen, indem Feedback als hilfreich oder nicht hilfreich bewertet wird oder es erfolgt eine Beschreibung in eigenen Worten.

3.1.2. Mein Kopfschmerz

Die App *Mein Kopfschmerz* [Pfi15] dient als Tagebuch zum Dokumentationszweck von Kopfschmerzen, um beispielsweise dem behandelnden Arzt einen Überblick über Auslöser der Kopfschmerzen, Symptome, Häufigkeit und Behandlung zu geben. Aus den Dateneingaben lässt sich ein Bericht erstellen, der dem Arzt vorgelegt werden kann [Pfi15]. In Bezug auf das TrackYourTinnitus-Projekt bietet diese App ein Beispiel für ein gelungenes Design, wie Abbildung 3.3 veranschaulicht. Der Schmerzgrad ist durch eine Skala-Einteilung (von 0-10) charakterisiert. Zusätzlich ist die Beeinträchtigung der täglichen Aktivitäten durch den Kopfschmerz durch farblich markierte Emotions-Bilder unterscheidbar. Die Nummerierung und farbliche Markierung unterstützen den Anwender bei der Bestimmung von Schmerz- und Beeinträchtigungsgrad. Diese Visualisierungsform lässt sich beispielsweise auch auf den App-Fragebogen von TYT übertragen. Zudem ist ebenso eine Kalender-Funktion möglich, durch die Tinnitus-Betroffene Therapie-Einträge erstellen und Therapie-Verläufe dokumentieren können.



Abbildung 3.3.: Mein Kopfschmerz [Pfi15]

3.1.3. CatchMyPain

Die App *CatchMyPain* [San15] dient als Schmerztagebuch, bei dem verschiedene Punkte, wie kurz- oder langfristige Schmerzangaben, Müdigkeit, Stress, Aktivitäten und Umstände, die Schmerzen verursachen bzw. Schmerzen verstärken erfasst werden. Mit der App lassen sich Schmerzaufzeichnungen anfertigen, um den Ort und die Intensität der Schmerzen zu definieren [San15]. Ein detaillierter Verlauf der persönlichen Schmerzen ist in einer Schmerzkurve einsehbar, wie Abbildung 3.4 zeigt.

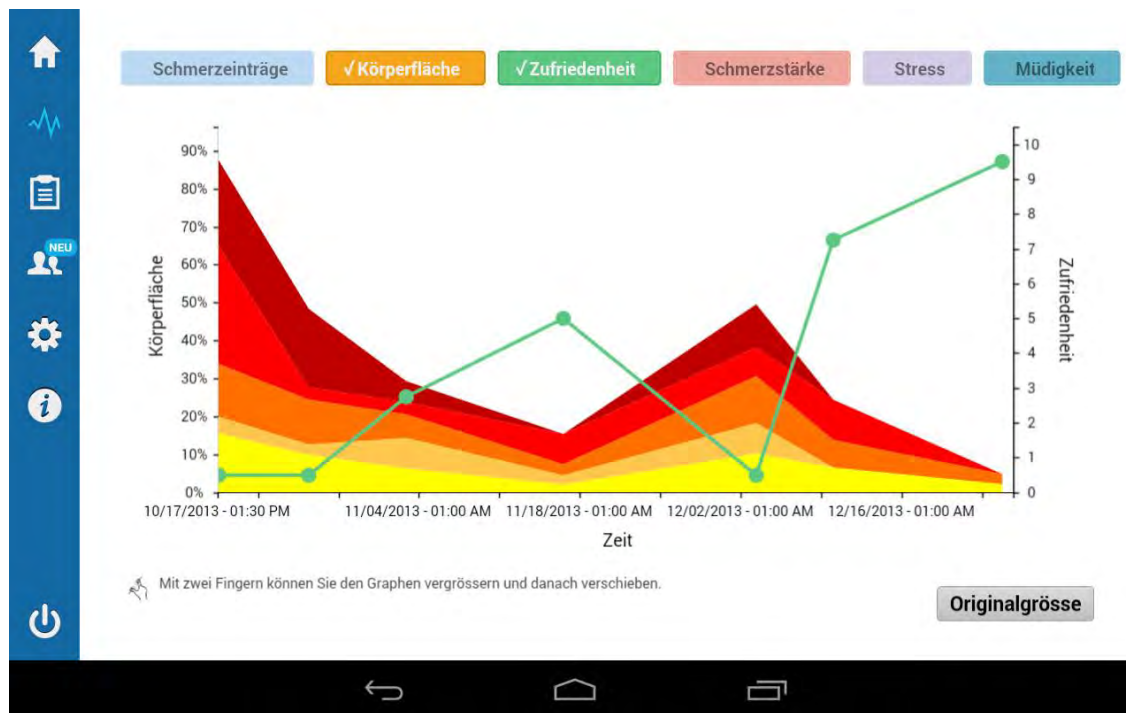


Abbildung 3.4.: CatchMyPain-Schmerzcurve [San15]

Zudem lassen sich die Statistiken exportieren, um sie Arzt oder Therapeuten zu zeigen. Ferner werden für einen Schmerzeintrag, wie in Abbildung 3.5 die Schmerzstärke, eine Beschreibung des Schmerzes, das Befinden, das Wetter sowie optional Kommentare erfasst [San15].

Die App bietet ein gutes Navigationskonzept und ein gelungenes Design. Alle Hauptmenüpunkte sind am linken Bildschirmrand zugänglich. Dies stellt sicher, dass sich Anwender innerhalb der App jederzeit orientieren können. Die App bietet eine Feedback-Funktionalität dahingehend, dass Anwender innerhalb der Gemeinschaft (Community) von CatchMyPain (vgl. Abbildung 3.6) selbst Feedback senden können. Weiter lassen sich über die Community z.B. Schmerz-Therapien austauschen. Der Austausch über eine Community ist auch für TrackYourTinnitus denkbar.

Anpassung eines Schmerzeintrags

Für die Wetterabfrage verwendete Zeit und Ort:

Zeitpunkt: 26.10.2014 - 21:40

Ort: Zurich, Switzerland

Wetter laden

Temperatur

-30 °C 8 °C 50 °C

Wolkenbedeckung

sonnig leicht bewölkt bewölkt bedeckt

Wind

windstill leicht mässig stark stürmisch Orkan

Schmerzstärke

Beschreibung
elektrisierend, heiss, stechend

Befinden
Zufriedenheit: 😞
Stress: nicht definiert
Müdigkeit: nicht definiert

Wetter
8 °C, bewölkt

Kommentar
Kein Kommentar vorhanden

Abbildung 3.5.: CatchMyPain-Schmerzeintrag [San15]

CatchMyPain Community

Thema erstellen

Schmerztherapie im Rheinland
von [User] 17 Beiträge 137 Zugriffe vor 14 Stunden

Rücken, linke Leiste starke Schmerzen
von [User] 11 Beiträge 205 Zugriffe vor einem Tag

Cluster Headake
von [User] 1 Beitrag 16 Zugriffe vor einem Tag

Crps mit Syringomyelie
von [User] 10 Beiträge 141 Zugriffe vor 2 Tagen

Arteritis temporalis
von [User] 1 Beitrag 17 Zugriffe vor 2 Tagen

migräne mit Aura
von [User] 9 Beiträge 150 Zugriffe vor 4 Tagen

Community Info

Freunde einladen

Feedback senden

Community Mitglieder

Abbildung 3.6.: CatchMyPain-Community [San15]

3.1.4. Metappbolic

Die App *metappbolic* [Aki14] dient Diabetis-Betroffenen zum Einsehen von Nährwerten aus einer Lebensmitteldatenbank, wie in Abbildung 3.7 veranschaulicht ist. Zudem steht ein Insulinrechner zur Verfügung, sodass ein ständiger Überblick über die persönlichen Vitalparameter gegeben ist. Weiter erhält der Anwender persönliches Feedback zu seiner Situation und Vorschläge zur Therapieverbesserung [Aki14]. Metappbolic dient nicht ausschließlich zu Dokumentationszwecken, sondern fokussiert sich auf Erfolge und eine einfache Integration der Therapie in den Alltag. In Form von gesetzten Herausforderungen und der Aussicht auf selbst gesetzte Belohnungen soll der Anwender motiviert bleiben. Die Erfolge lassen sich in sozialen Netzwerken teilen. Für TrackYourTinnitus können soziale Netzwerke dahingehend eingesetzt werden, dass sich Tinnitus-Betroffene mit anderen Betroffenen austauschen können.

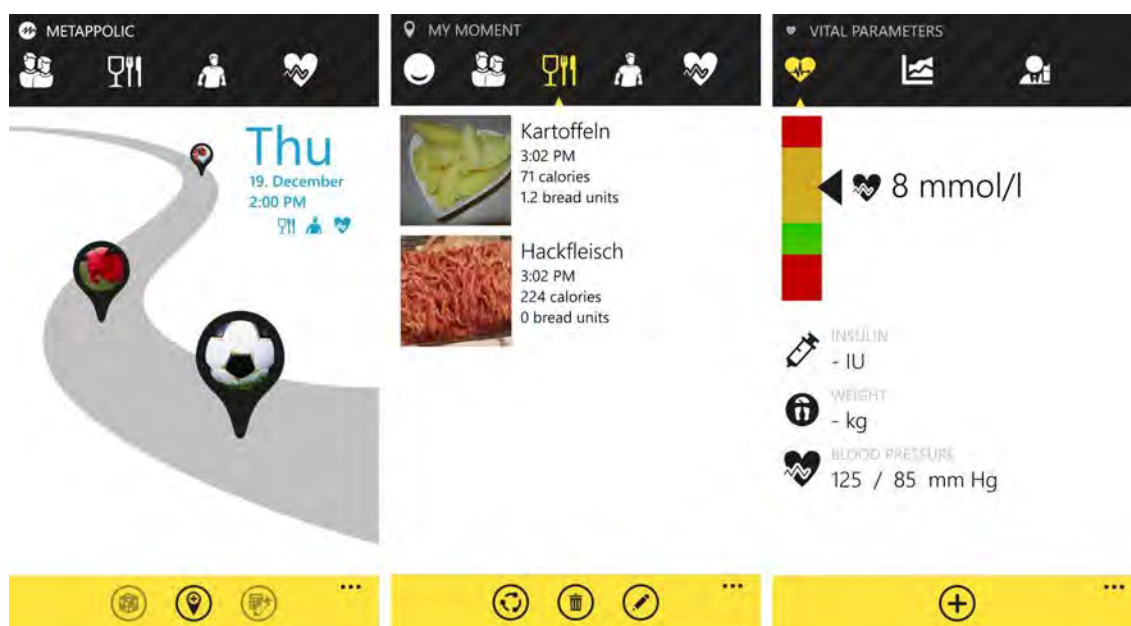


Abbildung 3.7.: metappbolic [Aki14]

3.2. Datenanalyse

Innerhalb der Datenanalyse wird der Ist-Zustand der Daten der TYT-Datenbank zu einem Tinnitus-Betroffenen betrachtet. Für diese Arbeit ist die Datenanalyse essentiell, da sie erforderlich ist, um Einsichten darüber zu erlangen, welche Feedback-Möglichkeiten auf Grundlage der vorhandenen Daten gegeben werden können. Zudem dient sie der Gewinnung von Erkenntnissen dahingehend, ob durch das Definieren von Regeln Feedback gegeben werden kann. Der Ist-Zustand der Daten ergibt sich aus dem App-Fragebogen und den drei statistischen Fragebögen: Mini Tinnitus Fragebogen (Mini-TF), Tinnitus Sample Case History Questionnaire (TSCHQ) und dem Fragebogen Schlimmstes Symptom [Sch13].

Der App-Fragebogen nimmt eine wesentliche Rolle ein, da er in regelmäßigen Abständen die Schwankungen der Tinnitus-Wahrnehmung der Betroffenen innerhalb der TYT-App erfasst. Alle Fragebögen und die zugehörigen Fragen sind im Anhang detailliert aufgeführt (vgl. Tabelle A.1 - A.4). Aus dem App-Fragebogen lassen sich Rückschlüsse bezüglich des Zusammenhangs zwischen der wahrgenommenen Tinnitus-Lautheit und verschiedenen Einflussfaktoren, wie z.B. Stress, Emotionen, Aufregung oder Konzentration treffen.

Der Mini Tinnitus Fragebogen stellt zwölf Fragen, die Rückschlüsse auf die Gefühle, das Verhalten und die Einstellung des Tinnitus-Betroffenen ermöglichen [HG04]. Die Fragen können dabei jeweils mit den Antwortmöglichkeiten "stimmt", "stimmt teilweise" oder "stimmt nicht" beantwortet werden. Der TSCHQ-Fragebogen bietet 35 verschiedene Fragen, die es ermöglichen sollen den Tinnitus besser einordnen zu können [Lan06]. Im Fragebogen zum schlimmsten Symptom [Sch13] gibt der Tinnitus-Betroffene aus einer Liste von Auswahloptionen an, welches er als für sich persönlich am schwerwiegendsten empfindet.

Zur Datenanalyse werden alle Fragen der drei statistischen Fragebögen in einer baumartigen Struktur erfasst, um die Möglichkeiten von Regeldefinitionen zu veranschaulichen. Zur graphischen Visualisierung eignet sich ein Entscheidungsbaum. Dabei handelt es sich um eine Repräsentationsform verzweigter Bedingungen [CL14]. Weiter lassen sich anhand von Entscheidungsbäumen leicht Regeln ableiten. Abbildung 3.8 zeigt einen Ausschnitt der erfassten Datenstruktur. Diese dient als Ausgangsbasis für das Konzept (vgl. Kapitel 4). Beispielsweise zielen die statistischen Fragebögen darauf ab herauszufinden, ob die Wahrnehmung des Tinnitus pulsierend ist und sofern dies zutrifft, ob der Tinnitus im Rhythmus des Herzschlags oder anders als der Herzschlag wahrgenommen wird. Weiter wird z.B. überprüft, ob der Tinnitus im linken Ohr, im rechten Ohr, in beiden Ohren, im Inneren des Kopfes oder an einer anderen Stelle wahrgenommen wird. Durch die unterschiedlichen Antwortoptionen lässt sich der entworfene Entscheidungsbaum der statistischen Fragebögen zum Definieren von Regeln und zur Bildung von Gruppen verwenden. Die Regeln bestimmen wann einem Tinnitus-Betroffenen bzw. wann einer definierten Gruppe Feedback gegeben wird. Tinnitus-Betroffene, die im linken Ohr einen pulsierenden Tinnitus im Rhythmus des Herzschlags wahrnehmen, können dadurch ein anderes Feedback erhalten, als Tinnitus-Betroffene, die den Tinnitus im Inneren des Kopfes und nicht-pulsierend wahrnehmen. Aus dem Entscheidungsbaum kann beispielsweise eine wie folgt definierte Regel abgelesen werden:

WENN *Tinnitus pulsierend* UND *im Rhythmus des Herzschlags* DANN *gib Feedback 1*.

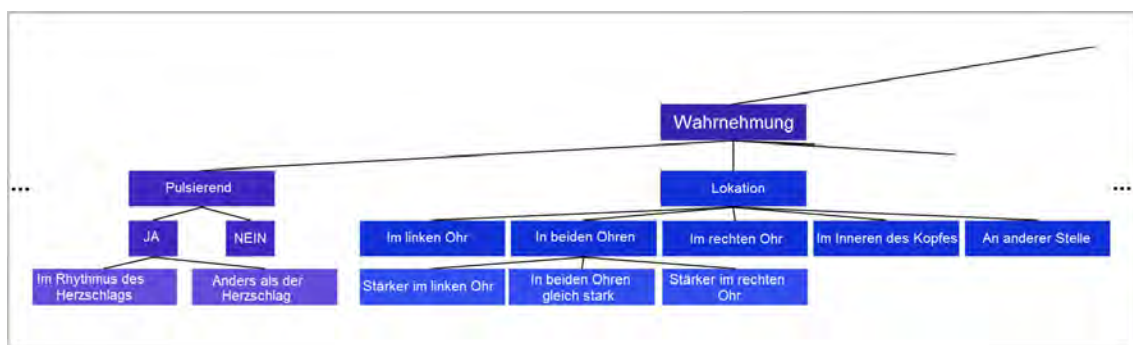


Abbildung 3.8.: Datenanalyse - Ausschnitt der Datenstruktur

3.3. Funktionsanalyse

Die Ermittlung möglicher Feedback-Funktionen, um die TrackYourTinnitus erweitert werden kann, erfolgt durch eine Funktionsanalyse in Form eines Szenarios und Anwendungsfalldiagrammen.

3.3.1. Szenario

Bei einem Szenario handelt es sich um ein konkretes Beispiel in Bezug auf das betrachtete System, das eine Folge von Interaktionsschritten beinhaltet und ein oder mehrere Ziele erfüllen soll [Poh07]. Das nachfolgende Szenario (vgl. Abbildung 3.9) veranschaulicht die Schritte der geplanten System-Erweiterung.

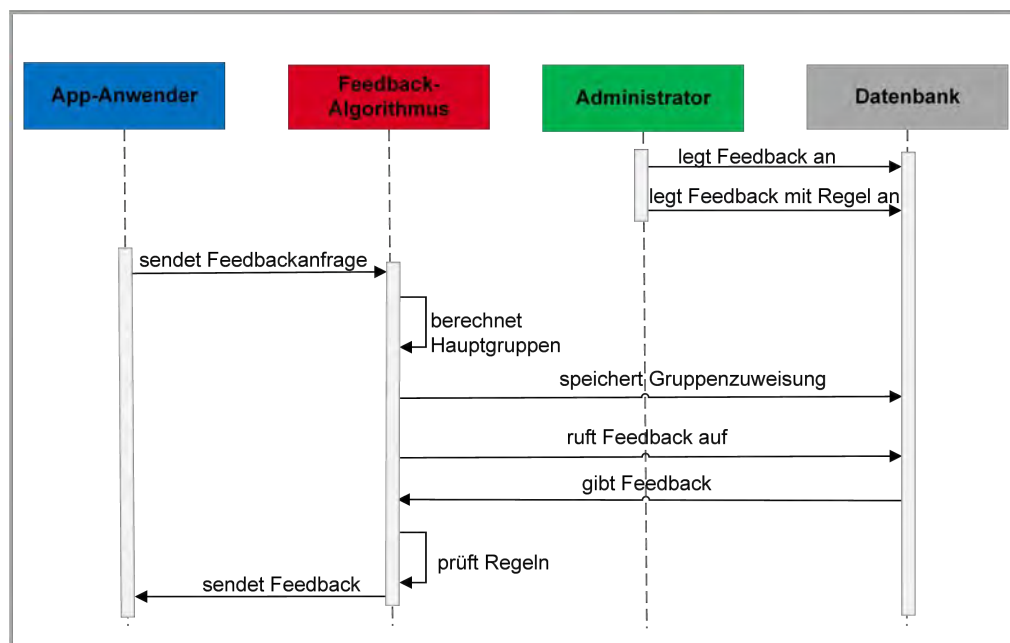


Abbildung 3.9.: Szenario - Ablauf Feedback-Algorithmus

Als Voraussetzung, um dem App-Anwender Feedbacks geben zu können, müssen Feedbacks angelegt sein. Zunächst legt ein Administrator Feedbacks an. Diese können für alle Tinnitus-Betroffene gültig sein. Zudem kann er Feedbacks mit Regeln anlegen. Diese werden dem App-Anwender lediglich dann gegeben, wenn die Regel für ihn zutrifft. Sobald der App-Anwender eine Feedbackanfrage sendet, berechnet der Feedback-Algorithmus zu welchen Gruppen ein App-Anwender auf Grundlage seiner ausgefüllten App-Fragebögen zugeordnet werden kann und es erfolgt eine Regelüberprüfung auf Basis der statistischen Fragebögen (vgl. Kapitel 3.2). Sind diese Schritte abgeschlossen, erhält der App-Anwender ein zutreffendes Feedback.

3.3.2. Anwendungsfalldiagramme

Anwendungsfalldiagramme beschreiben die Funktionalitäten des Systems bzw. der einzelnen Systemkomponenten aus Benutzersicht [Par10]. Dazu werden Akteure benannt und mit den Anwendungsfällen (engl. Use Cases), die sie betreffen verbunden. Anwendungsfalldiagramme eignen sich dazu, die Beziehung der Anwendungsfälle von Systemkomponenten mit den Akteuren darzustellen [Poh07].

Zu den Akteuren, die das System verwenden zählen Administratoren und Tinnitus-Betroffene, die im Folgenden als App-Anwender bezeichnet werden. Dabei handelt es sich um Tinnitus-Betroffene, die die Android-App verwenden. Administratoren tragen dafür Sorge, dass für eine Vielzahl an App-Anwender ausreichend Feedbacks vorhanden sind. Abbildung 3.10 veranschaulicht die Anwendungsfälle, die aus Sichtweise des Administrators an das zu realisierende System gestellt werden.

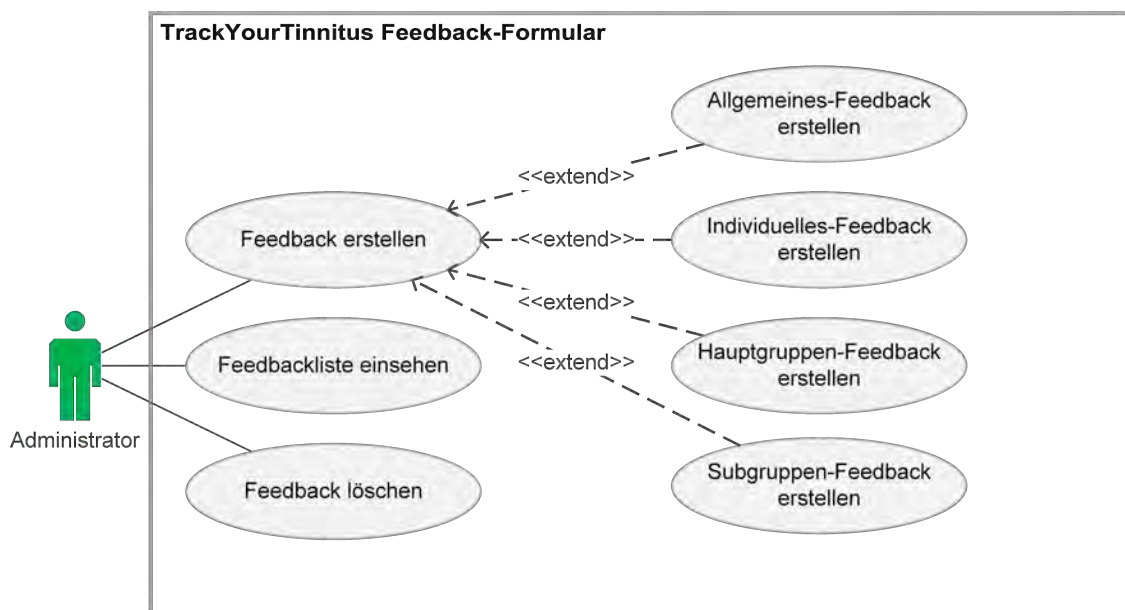


Abbildung 3.10.: Anwendungsfalldiagramm - Administrator

Ein Administrator muss über ein Feedback-Formular im Adminbereich der TYT-Web-Anwendung in der Lage sein, Feedback zu erstellen, eine Feedbackliste aller verfügbaren Feedbacks einzusehen und Feedback bei Bedarf zu löschen. Der Anwendungsfall *Feedback erstellen* wird um jeweils einen der Anwendungsfälle *Allgemeines-Feedback erstellen*, *Individuelles-Feedback erstellen*, *Hauptgruppen-Feedback erstellen* oder *Subgruppen-Feedback erstellen* erweitert. Eine Definition dieser Feedback-Arten wird in Kapitel 4.3 gegeben.

Die Anwendungsfälle des App-Anwenders sind in Abbildung 3.11 dargestellt. Der App-Anwender muss Feedbacks zu den Einflussfaktoren einsehen können, zu den einzelnen Feedbacks eine Detailbeschreibung ansehen können und das erhaltene Feedback dahingehend bewerten, ob es für ihn hilfreich oder nicht hilfreich ist. Zudem kann er die Datenzuverlässigkeit (vgl. Kapitel 4.4) einsehen. Die Datenzuverlässigkeit soll farblich kennzeichnen, ob die vorhandenen

Daten ausreichend sind, um dem App-Anwender verlässliche Feedbacks zu geben. Zudem kann er den Datumsbereich ändern, für den die Berechnung der Gruppen stattfinden soll. Hierdurch können beispielsweise größere Zeiträume miteinbezogen werden und es kann zurückverfolgt werden, welche Einflussfaktoren zu einem früheren Zeitraum einen Einfluss auf den Tinnitus gehabt haben. Ferner kann der App-Anwender jede Woche einen neuen Tipp der Woche einsehen, wodurch er fortlaufend und aktuell Feedbacks erhält, die ihm im Umgang mit seinem Tinnitus helfen können. Der App-Anwender kann selbst in Form von Text Feedback geben, um mitzuteilen, welche Maßnahmen bzw. Therapien ihm zur Linderung des Tinnitus helfen.

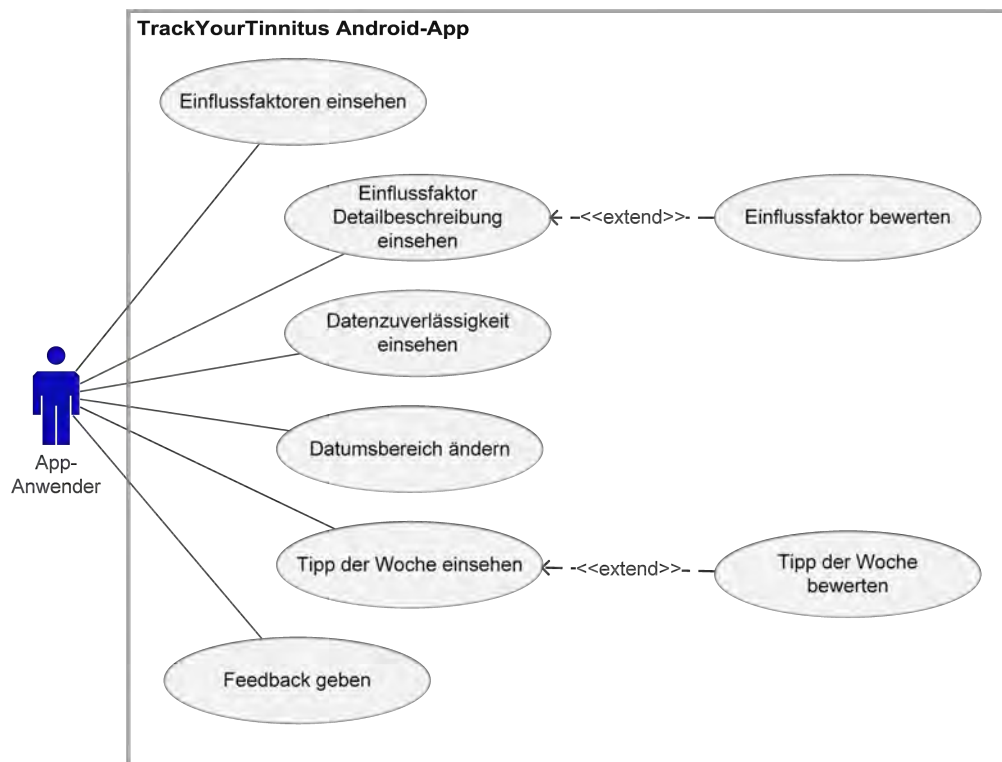


Abbildung 3.11.: Anwendungsfalldiagramm - Anwender

3.4. Anforderungsdefinition

Alle an die Arbeit gestellten Anforderungen sind als funktionale und nichtfunktionale Anforderungen nachfolgend definiert. Die Anforderungen sind neben der Bezeichnung und Beschreibung durch eine eindeutige Identifikationsnummer (ID) und dem Kriterium (Kr.) beschrieben. Die Kriterien *Muss(M)* oder *Kann(K)* legen fest, wie der Erfüllungsgrad einer Anforderung zu realisieren ist. Muss-Anforderungen müssen im Konzept berücksichtigt und implementiert sein. Kann-Anforderungen müssen konzipiert, aber nicht zwingend implementiert sein.

3.4.1. Funktionale Anforderungen

Die nachfolgende Tabelle 3.1 bildet alle definierten funktionalen Anforderungen ab. Eine funktionale Anforderung (FA) beschreibt die gewünschte Funktionalität an das System bzw. an die Systemkomponenten [Poh07]. Hierbei werden die drei Komponenten Adminbereich, Android-Applikation und Feedback-Algorithmus betrachtet.

Tabelle 3.1.: Funktionale Anforderungen

ID	Bezeichnung	Beschreibung	Kr.
Adminbereich			
FA-01	Allgemeines-Feedback erstellen	Der Administrator muss Feedback, das für alle Tinnitus-Betroffene allgemein gültig ist in Deutsch und Englisch erstellen und speichern können.	M
FA-02	Individuelles-Feedback erstellen	Der Administrator muss Feedback, das für Tinnitus-Betroffene individuell auf Basis der gegebenen Antworten zu den Fragebögen gültig ist in Deutsch und Englisch erstellen und speichern können.	M
FA-03	Hauptgruppen-Feedback erstellen	Der Administrator muss zu den Tinnitus-Hauptgruppen, in die Tinnitus-Betroffene eingeordnet werden, die zugehörigen Feedbacks in Deutsch und Englisch erstellen und speichern können.	M
FA-04	Subgruppen-Feedback erstellen	Der Administrator muss für Hauptgruppen Subgruppen-Feedbacks in Deutsch und Englisch erstellen und speichern können.	M
FA-05	Regeln erstellen	Der Administrator muss Regeln erstellen können, die festlegen unter welchen Bedingungen welches Feedback gegeben wird.	M
FA-06	Feedbacks einsehen	Der Administrator kann erstellte Feedbacks in Form einer Liste einsehen.	K
FA-07	Feedbacks löschen	Der Administrator kann erstellte Feedbacks löschen.	K
Android-Applikation			
FA-08	Allgemeines-Feedback erhalten	Der App-Anwender muss basierend auf seinen Eingabedaten Allgemeines-Feedback erhalten können, das generell zutrifft. Dieses Feedback ist keiner speziellen Hauptgruppe zugewiesen.	M
FA-09	Hauptgruppen-Feedback erhalten	Der App-Anwender muss basierend auf seinen Eingabedaten Feedback erhalten, das für seine Hauptgruppe gilt. Hauptgruppen-Feedbacks werden in der Ansicht "Einflussfaktoren" angezeigt.	M
FA-10	Subgruppen-Feedback erhalten	Der App-Anwender muss basierend auf seinen Eingabedaten individuelles Feedback erhalten können, das für seine Subgruppe gilt.	M
FA-11	Tipp der Woche erhalten	Der App-Anwender muss jede Woche einen neuen Tipp der Woche abrufen können. Als Tipp der Woche lassen sich alle Feedback-Arten anzeigen.	M

FA-12	Einflussfaktoren sortiert einsehen	Der App-Anwender muss das erhaltene Hauptgruppen-Feedback der Einflussfaktoren sortiert einsehen können. Dies erfolgt, indem bei der Anzeige die berechnete Datenzuverlässigkeit berücksichtigt wird.	M
FA-13	Datumsbereich ändern	Der App-Anwender muss den Datumsbereich zur Berechnung der Hauptgruppen-Feedbacks in der Einflussfaktoren-Ansicht ändern können.	M
FA-14	Feedback bewerten	Der App-Anwender muss alle erhaltenen Feedbacks, in der Einflussfaktoren-Ansicht sowie Feedbacks vom Tipp der Woche bewerten können und hierdurch Rückmeldung darüber geben, ob das Feedback hilfreich war.	M
FA-15	Feedback geben	Der App-Anwender kann selbst Feedback über die App geben. Er hat die Option Feedback per E-Mail zu geben, das ihm im Umgang mit dem Tinnitus hilft.	K
FA-16	Feedback speichern	Geladene Feedbacks können von der Android-Applikation gespeichert werden. Dadurch kann der App-Anwender das zuletzt erhaltene Feedback auch dann einsehen, wenn er über keine Internetverbindung verfügt.	K
Feedback-Algorithmus			
FA-17	Hauptgruppen ermitteln	Basierend auf den Antworten zu den App-Fragebögen eines App-Anwenders, muss der Hauptgruppen-Algorithmus den App-Anwender den richtigen Hauptgruppen zuweisen.	M
FA-18	Regeln auswerten	Die Regeln, die der Administrator definiert, müssen vom Feedback-Algorithmus ausgewertet werden, um Subgruppen-Feedbacks zu ermitteln.	M
FA-19	Datenzuverlässigkeit berechnen	Zur Aussage, ob die vorhandenen Datensätze der App-Fragebögen zuverlässig sind, muss die Datenzuverlässigkeit (Degree of Reliability - DOR) berechnet werden.	M
FA-20	Flags setzen	Datensätze, die zur Berechnung der Hauptgruppen eines App-Anwenders verwendet werden, können in der Datenbank als verarbeitete Daten mit einem Flag versehen werden.	K
FA-21	Statistische Fragebögen vollständig bereitstellen	Zur korrekten Bereitstellung aller Fragen der statistischen Fragebögen für den Adminbereich, müssen alle Fragen in der Datenbank gespeichert sein. Die Datenbank muss um die Fragen 15-35 des TSCHQ ergänzt werden.	M

3.4.2. Nichtfunktionale Anforderungen

Tabelle 3.1 bildet alle definierten nichtfunktionalen Anforderungen ab. Eine nichtfunktionale Anforderung (NFA) verdeutlicht die Qualität, in der die Funktionalität erbracht werden muss [Poh07]. Dabei werden nichtfunktionale Anforderungen für die Datenbereitstellung, -verarbeitung und -speicherung sowie für die grafische Benutzeroberfläche definiert.

Tabelle 3.2.: Nichtfunktionale Anforderungen

ID	Bezeichnung	Beschreibung	Kr.
Datenbereitstellung, -verarbeitung und -speicherung			
NFA-01	Hauptgruppen- definition	Zur Auswahl von Hauptgruppen im Adminbereich und zur Zuweisung eines App-Anwenders zu einer Hauptgruppe im Hauptgruppen-Algorithmus, müssen die Hauptgruppen im Vorfeld definiert und in der Datenbank angelegt sein.	M
NFA-02	Verarbeitung korrekter App-Datensätze	Ist ein App-Fragebogen unvollständig ausgefüllt, ist der zugehörige Datensatz in der Datenbank für den Hauptgruppen-Algorithmus unbrauchbar und muss durch Herausfiltern berücksichtigt werden.	M
NFA-03	Internationalisierung	Alle Feedbacks, die der Administrator anlegt, müssen in Deutsch und Englisch speicherbar sein.	M
NFA-04	Datenspeicherung der Hauptgruppen	Die Zuweisung eines App-Anwenders zu den Hauptgruppen muss in der Datenbank in einem Datensatz gespeichert werden, nachdem der Hauptgruppen-Algorithmus durchgeführt wird.	M
NFA-05	Korrekte Datenhaltung	Zur korrekten Bereitstellung der Fragen der statistischen Fragebögen, muss die Abfrage der Pflichtfragen bzw. optionalen Fragen richtig angepasst werden.	M
Grafische Benutzeroberfläche			
NFA-06	Listen-basierte Feedback- Darstellung	Feedbacks für die Einflussfaktoren-Ansicht müssen als Liste dargestellt werden.	M
NFA-07	Detaillierte Feedback-Darstellung	Feedbacks für den Tipp der Woche und einzeln selektierte Feedbacks der Einflussfaktoren-Ansicht müssen detailliert dargestellt werden.	M
NFA-08	Datenzuverlässigkeit grafisch darstellen	Die Zuverlässigkeit der Daten muss durch den berechneten DOR-Wert grafisch dargestellt werden: zuverlässig (grün), einigermaßen zuverlässig (gelb), nicht zuverlässig (grau).	M
NFA-09	Design	Die Darstellung des Feedback-Formulars im Adminbereich sowie der App-Erweiterung kann sich am bestehenden Design von TYT orientieren.	K
NFA-10	Mehrsprachigkeit der Feedbacks	Die Feedbacks sollen dem App-Anwender in der richtigen Sprache angezeigt werden.	M

4

Konzept

In diesem Kapitel wird das Konzept erläutert, das der Implementierung eines Feedback-Algorithmus dient (vgl. Kapitel 5). Kapitel 4.1 beschreibt vier Umsetzungsoptionen, die die Realisierung eines Feedback-Algorithmus ermöglichen und begründet die Wahl, welche Option realisiert wird. Kapitel 4.2 beschreibt, wie Tinnitus-Betroffene gruppiert werden können, um für diese Gruppen ein Feedback geben zu können. Auf Basis der Antworten zu den statistischen Fragebögen und dem App-Fragebogen definiert Kapitel 4.3 verschiedene Feedback-Arten, die Tinnitus-Betroffenen geben werden können. Zudem stellt Kapitel 4.4 das entwickelte Konzept des Feedback-Algorithmus vor und beschreibt, wie die Daten für jeden App-Anwender verarbeitet werden, um ihn verschiedenen Tinnitus-Gruppen zuzuweisen. Kapitel 4.5 erläutert das Usability-Konzept, das es Administratoren erlaubt Feedbacks anzulegen und zu definieren, für welche Tinnitus-Gruppen die Feedbacks gültig sind. Weiter wird das Navigationskonzept der App vorgestellt sowie die Darstellung der unterschiedlichen Feedback-Arten in der App.

4.1. Umsetzungsoptionen

Zur Realisierung des Feedback-Algorithmus werden mögliche Umsetzungsoptionen dahingehend vorgestellt, wie Tinnitus-Gruppen auf unterschiedliche Weise gebildet werden können und wie Tinnitus-Betroffenen Feedback gegeben wird. Anhand einer dieser Optionen wird in den darauffolgenden Kapiteln ein detailliertes Konzept für die Implementierung erstellt.

Option 1

Durch die Daten, die zu einem Tinnitus-Betroffenen auf Basis der beantworteten Fragebögen vorhanden sind, wird ein Entscheidungsbaum basierend auf der Datenanalyse (vgl. Kapitel 3.2) gebildet. Der Algorithmus ermittelt anhand des Entscheidungsbaumes Tinnitus-Gruppen und weist den Tinnitus-Betroffenen einer Gruppe zu. Hierzu wird jede Antwortoption der Fragebögen bei mehreren möglichen Antworten, vorab mit einem Wert belegt. Füllt ein Tinnitus-Betroffener alle Fragen der Fragebögen aus, berechnet sich durch die vordefinierten Werte ein Gesamtwert. Dieser errechnete Wert, der sich durch alle Pfade des Entscheidungsbaums ergibt, entscheidet welcher Gruppe ein Tinnitus-Betroffener zuzuteilen ist. Hierdurch kann definiert werden, welches Feedback eine Tinnitus-Gruppe erhält. Außerdem existieren zusätzliche Regeln, die bestimmen welches Feedback für welche Gruppe gegeben wird.

Option 2

Zusätzlich zu Option 1 lässt sich zur Gruppenbildung eine Gewichtung bewirken, hinsichtlich Einflussfaktoren, wie beispielsweise: Stress, Emotionen, Konzentration und Aufregung. Sofern

verschiedene Fragen aus den Fragebögen auf denselben Einflussfaktor schließen lassen und ein Betroffener alle diese Fragen als zutreffend beantwortet (z.B. 5 von 5 zutreffende Fragen für den Einflussfaktor Emotion), lässt dies Rückschlüsse dahingehend zu, dass dieser Einflussfaktor einen hohen Einfluss auf die Tinnitus-Belastung des Tinnitus-Betroffenen hat (z.B. sehr hohe emotionale Belastung).

Option 3

Im Gegensatz zu Option 1 und Option 2 können die Tinnitus-Gruppen anhand des Schweregrades des Tinnitus durch Auswertung des Mini-Tinnitus-Fragebogen festgelegt werden. Demnach existieren für Schweregrad 1-4 jeweils eine Gruppe. Anschließend erfolgt das Feedback durch definierte Regeln für jede Gruppe, die basierend auf Alter, Geschlecht und weiteren frei wählbaren Bedingungen aus dem Entscheidungsbaum definiert werden können.

Option 4

Eine andere Möglichkeit Gruppen zu bilden, besteht in der Berechnung von Korrelationskoeffizienten, zwischen der Tinnitus-Lautheit und den Einflussfaktoren: Stress, Emotionen, Konzentration und Aufregung. Die Werte zur Berechnung des Korrelationskoeffizienten beziehen sich aus den beantworteten App-Fragebögen eines Tinnitus-Betroffenen. Die Gruppeneinteilung erfolgt über den berechneten Korrelationskoeffizienten jedes einzelnen Einflussfaktors. Für jede daraus ermittelte Hauptgruppe lassen sich Subgruppen bilden, indem Regeln definiert werden. Als Regeln lassen sich die Fragen und Antwortoptionen der statistischen Fragebögen heranziehen (z.B. wenn der Tinnitus-Betroffene Frage 8 des Mini-TF mit "stimmt" beantwortet hat, dann gib Feedback 2).

Zur Realisierung des Feedback-Algorithmus wird Option 4 umgesetzt. Diese Option bietet im Gegensatz zu den anderen vorgestellten Optionen den Vorteil, dass die Berechnung von Korrelationskoeffizienten aus der Tinnitus-Lautheit und den Einflussfaktoren einen genauen Zusammenhang zwischen dem Tinnitus des Betroffenen und den jeweiligen Einflussfaktoren zeigt und damit eine Gruppeneinteilung ermöglicht. Die Gruppeneinteilung veranschaulicht zum einen, ob es einen Zusammenhang zu verschiedenen Einflussfaktoren gibt und erlaubt es zum anderen ein Feedback einmalig zu erstellen, das anschließend einer Vielzahl an Tinnitus-Betroffenen gegeben wird. Weiter bietet diese Option den Vorteil, dass durch den Entscheidungsbaum leicht Regeln abgeleitet werden können und sich dadurch Subgruppen bilden lassen.

4.2. Hauptgruppendefinition

Dieses Unterkapitel erarbeitet ein Konzept, das Tinnitus-Betroffene verschiedenen Gruppen zuordnet, für die der zu entwickelnde Feedback-Algorithmus Feedbacks geben soll. Betrachtet werden die vier Einflussfaktoren: Stress, Emotion, Konzentration und Aufregung. Je Einflussfaktor wird der Betroffene einer von drei Gruppen zugewiesen und insgesamt befindet er sich in vier von zwölf möglichen Hauptgruppen.

4.2.1. Korrelationskoeffizienten

Die Hauptgruppen werden ermittelt, indem die Zusammenhänge zwischen der Tinnitus-Lautheit und jeweils einem Einflussfaktor untersucht wird. Zur Untersuchung der

Zusammenhänge wird eine Korrelation berechnet, die aus Korrelationskoeffizienten r und p-Wert besteht. Der Korrelationskoeffizient bestimmt die Stärke, sowie die Richtung des Zusammenhanges und ist durch einen Wert zwischen -1 und 1 repräsentiert. Liegt der Wert nahe 1 kennzeichnet dies einen starken positiven Zusammenhang. Ein Wert nahe -1 spricht für einen starken negativen Zusammenhang. Liegt der Wert bei etwa 0 besteht kein Zusammenhang. Zu unterscheiden sind der positive- und negative Korrelationskoeffizient [Hü06]. Die Unterschiede werden am Beispiel des Zusammenhanges zwischen der Tinnitus-Lautheit und dem Einflussfaktor Stress erläutert.

Positiver Korrelationskoeffizient:

- Große x-Werte sind mit und großen y-Werten verbunden: hohe Tinnitus-Lautheit ist mit hohem Stress verbunden.
- Bzw. kleine x-Werte sind mit kleinen y-Werten verbunden: geringe Tinnitus-Lautheit ist mit geringem Stress verbunden.

Negativer Korrelationskoeffizient:

- Kleine x-Werte sind mit großen y-Werten verbunden: geringe Tinnitus-Lautheit bei hohem Stress.
- Bzw. große x-Werte sind mit kleinen y-Werten verbunden: hohe Tinnitus-Lautheit trotz geringem Stress.

Formel 4.1 stellt die mathematische Formel zur Berechnung des Korrelationskoeffizienten nach Pearson [Hü06] vor. Dieser Korrelationskoeffizient gilt nur für lineare Zusammenhänge, dabei handelt es sich um Zusammenhänge, die im Streudiagramm durch eine Gerade abgebildet werden. Hierbei steht das Merkmal x für den Wert der Tinnitus-Lautheit und das Merkmal y für den Wert eines Einflussfaktors, wie z.B. Stress. Der Index i kennzeichnet die Identifikationsnummer des App-Fragebogens. Bei \bar{x} handelt es sich um den Mittelwert aller Werte zur Tinnitus-Lautheit bzw. bei \bar{y} um den Mittelwert des jeweiligen Einflussfaktors. Die Formel 4.1 berechnet im Zähler zunächst die Summe aller Produkte zwischen den x-Werten und y-Werten, abzüglich ihrer Mittelwerte. Im Nenner werden die Summen der x-Werte und y-Werte getrennt gebildet. Dabei wird von den Werten vor der Summierung der entsprechende Mittelwert abgezogen und das Ergebnis quadriert. Die Wurzeln der so gebildeten Summen werden anschließend multipliziert. Das Ergebnis der Division von Zähler und Nenner ist der Korrelationskoeffizient r .

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (4.1)$$

Der p-Wert zeigt, ob sich der Korrelationskoeffizient signifikant von 0 unterscheidet, indem er die Nullhypothese¹ widerlegt [Hü06]. Dies ist bei einem p-Wert kleiner als 0.05 der Fall. Sofern die Korrelation statistisch signifikant ist, schließt dies darauf, dass sie auch bedeutsam ist und

¹Die Nullhypothese bezeichnet in der Statistik eine Behauptung, die es zu widerlegen gilt [Hü06]. Beispielsweise, dass der Korrelationskoeffizient nicht signifikant ist.

dem Tinnitus-Betroffenen kann eine Rückmeldung gegeben werden. Ist die Korrelation nicht signifikant, sind zu wenig Daten vorhanden oder sie ist nicht bedeutsam.

4.2.2. Einflussfaktoren

Nachfolgend werden Schaubilder einzelner Einflussfaktoren sowie die Gruppeneinteilung gezeigt. Zusätzlich veranschaulicht Tabelle 4.1 die Hauptgruppendefinitionen. Für jeden Einflussfaktor existiert jeweils eine Gruppe für Betroffene, die eine positive Korrelation aufweisen und eine Gruppe für Betroffene, die eine negative Korrelation aufweisen. Um den Fall abzudecken, dass ein Tinnitus-Betroffener weder eine positive noch eine negative Korrelation aufweist, gibt es je Einflussfaktor eine weitere Gruppe, die alle Tinnitus-Betroffene beinhaltet, bei denen kein Zusammenhang zum Einflussfaktor existiert.

Tabelle 4.1.: Hauptgruppendefinitionen

Hauptgruppe	Beschreibung	Wert
Einflussfaktor Stress		
Gruppe 1	Positive Stress-Korrelation	$r \geq 0.7$
Gruppe 2	Negative Stress-Korrelation	$r \leq -0.7$
Gruppe 3	Zusammenhangslos: Tinnitus-Lautheit & Stress	$0.7 > r > -0.7$
Einflussfaktor Konzentration		
Gruppe 4	Positive Konzentration-Korrelation	$r \geq 0.7$
Gruppe 5	Negative Konzentration-Korrelation	$r \leq -0.7$
Gruppe 6	Zusammenhangslos: Tinnitus-Lautheit & Konzentration	$0.7 > r > -0.7$
Einflussfaktor Emotion		
Gruppe 7	Positive Emotion-Korrelation	$r \geq 0.7$
Gruppe 8	Negative Emotion-Korrelation	$r \leq -0.7$
Gruppe 9	Zusammenhangslos: Tinnitus-Lautheit & Emotion	$0.7 > r > -0.7$
Einflussfaktor Aufregung		
Gruppe 10	Positive Aufregung-Korrelation	$r \geq 0.7$
Gruppe 11	Negative Aufregung-Korrelation	$r \leq -0.7$
Gruppe 12	Zusammenhangslos: Tinnitus-Lautheit & Aufregung	$0.7 > r > -0.7$

Abbildung 4.1 zeigt sortiert alle Tinnitus-Betroffene ("sorted User") mit ihren Korrelationskoeffizienten aus dem Zusammenhang Tinnitus-Lautheit und Stress an. Hierbei fallen alle Betroffenen, die einen Korrelationskoeffizienten größer oder gleich 0.7 haben in die Gruppe 1 (Positive Stress-Korrelation) und alle, die kleiner 0.7 und gleichzeitig größer -0.7 sind in Gruppe 3 (Zusammenhangslos: Tinnitus-Lautheit & Stress). In diesem Beispiel gibt es keine Betroffenen, die Gruppe 2 (Negative Stress-Korrelation) zugeordnet werden können. Der p-Wert aller Betroffenen zu der Tinnitus-Lautheit-Stress-Korrelation ist in Abbildung 4.2 dargestellt. Hier ist zu erkennen, dass sich ca. 60% der Betroffenen in dem Bereich befinden, in dem signifikante Werte ermittelt werden.

In Abbildung 4.3 ist der Zusammenhang zwischen Tinnitus-Lautheit und Konzentration der Tinnitus-Betroffenen auf dieselbe Weise, wie in Abbildung 4.1, dargestellt. Mit dem Unterschied, dass hier Betroffene in der Gruppe "Negative Konzentration-Korrelation" vorkommen.

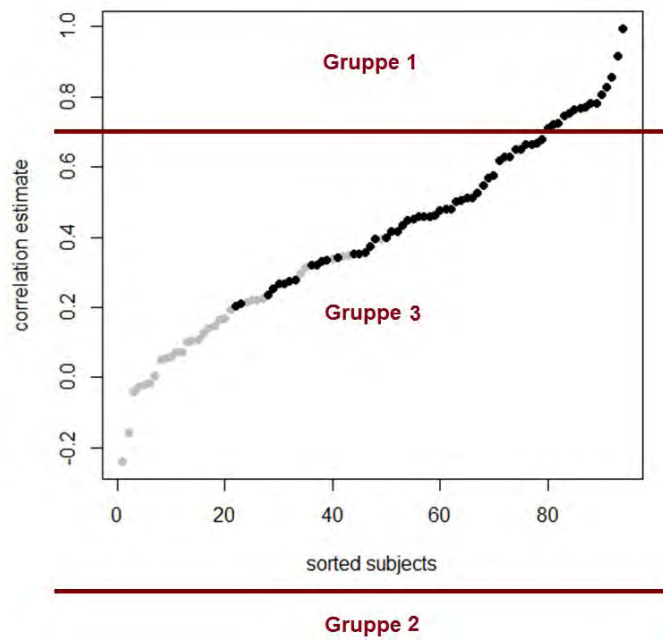


Abbildung 4.1.: Tinnitus-Gruppen - Einflussfaktor Stress

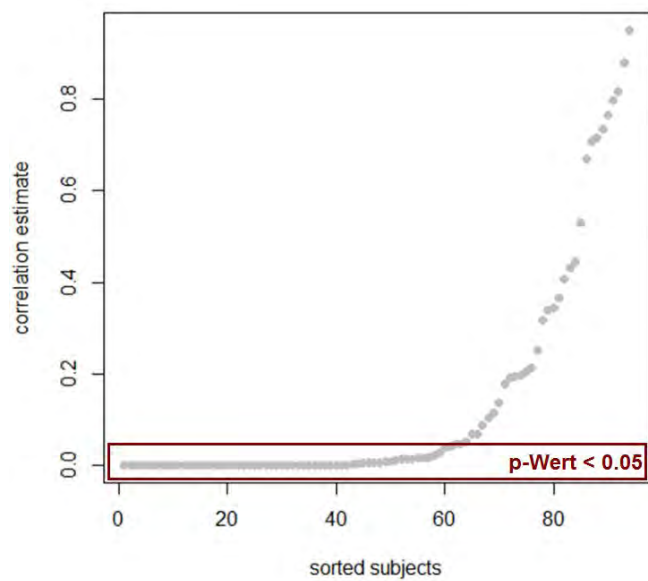


Abbildung 4.2.: p-Wert - Einflussfaktor Stress

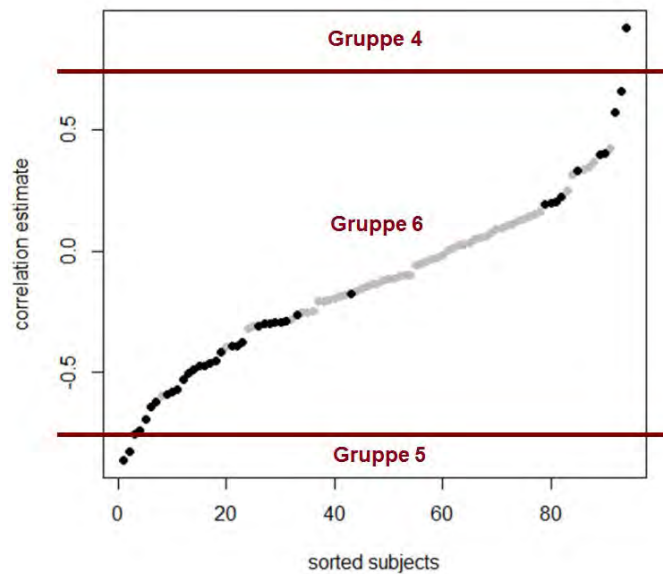


Abbildung 4.3.: Tinnitus-Gruppen - Einflussfaktor Konzentration

4.3. Feedback-Arten

Es werden vier verschiedene Feedback-Arten unterschieden, die ein App-Anwender erhalten und ein Administrator über ein Feedback-Formular auf der TrackYourTinnitus-Webseite eingeben kann. Im weiteren Verlauf werden die verschiedenen Feedback-Arten näher beschrieben.

1. Allgemeines-Feedback

Diese Feedback-Art ist allgemein für alle App-Anwender aller Gruppen gültig und ohne Regel definiert. Beispielsweise kann es dazu verwendet werden, um allen App-Anwendern neue wissenschaftliche Erkenntnisse zu übermitteln, die als Tipp der Woche (vgl. Kapitel 3.3) in der App angezeigt werden.

2. Individuelles-Feedback

Individuelles-Feedback wird durch die Definition einer Regel bestimmt und ist für alle App-Anwender gültig, für die die definierte Regel zutrifft, unabhängig welchen Gruppen ein App-Anwender zugeordnet ist. Im Gegensatz zum allgemeinen Feedback ist es jedoch nicht für alle App-Anwender gültig, sondern kann unabhängig der Hauptgruppen, in denen sich Tinnitus-Betroffene befinden als Tipp der Woche gegeben werden. Als Regel kann beispielsweise festgelegt sein, dass das Feedback allen Tinnitus-Betroffenen gegeben wird, die den Tinnitus im linken Ohr wahrnehmen.

3. Hauptgruppen-Feedback

Dieses Feedback ist für eine bestimmte Hauptgruppe definiert und ist für alle Tinnitus-Betroffene, die dieser Gruppe zugewiesen sind gültig, z.B. Tinnitus-Betroffene mit einer positiven Korrelation zwischen der Tinnitus-Lautheit und Stress (Gruppe 1).

4. Subgruppen-Feedback

Das Subgruppen-Feedback ermöglicht eine genauere Unterteilung der Hauptgruppen,

indem zusätzlich zu den Hauptgruppen Regeln definiert werden, z.B. das Feedback ist für alle Tinnitus-Betroffenen gültig, die Gruppe 1 zugeordnet sind und den Tinnitus im linken Ohr wahrnehmen.

Tabelle 4.2 listet die verschiedenen Feedback-Arten auf und zeigt, ob eine Feedback-Art einer einzelnen Gruppe zugewiesen ist oder für alle Gruppen gültig ist und ob die Definition von Regeln erforderlich ist.

Tabelle 4.2.: Feedback-Arten

Feedback-Art	Gruppenzuweisung	Regeldefinition
Allgemeines Feedback	Alle Gruppen	Ohne Regel
Individuelles-Feedback	Alle Gruppen	Mit Regel
Hauptgruppen-Feedback	Einzelne Gruppe	Ohne Regel
Subgruppen-Feedback	Einzelne Gruppe	Mit Regel

4.4. Feedback-Algorithmus

Nachfolgend wird ein Konzept definiert, das es ermöglicht einem App-Anwender Feedback zu geben, das ihn dabei unterstützen soll die Beschwerden seines Tinnitus zu lindern. Das Konzept des Feedback-Algorithmus besteht aus einem Hauptgruppen-Algorithmus, der den App-Anwender anhand der ausgefüllten App-Fragebögen verschiedenen Hauptgruppen zuordnet (vgl. Kapitel 4.2) und anschließend die zugehörigen Hauptgruppen-Feedbacks ermittelt. Zudem besteht der Feedback-Algorithmus aus einer Regelauswertung, die für einen App-Anwender überprüft, ob für ihn Individuelles-Feedback bzw. Subgruppen-Feedback zutrifft. Die Regelauswertung beruht auf den statistischen Fragebögen (vgl. Kapitel 3.2).

4.4.1. Hauptgruppen-Algorithmus

Dieses Unterkapitel beschreibt das Konzept des Hauptgruppen-Algorithmus. Der Hauptgruppen-Algorithmus wertet für jeden App-Anwender einzeln die eingegebenen Daten aus dem App-Fragebogen aus und weist ihn den definierten Hauptgruppen (vgl. Tabelle 4.1) zu, indem der Zusammenhang zwischen der Tinnitus-Lautheit und den einzelnen Einflussfaktoren (Stress, Aufregung, Emotion und Konzentration) berechnet wird. Dies erfolgt durch die Berechnung des entsprechenden Korrelationskoeffizienten. Der Ablauf des Hauptgruppen-Algorithmus ist durch die folgenden Schritte beschrieben:

Schritt 1: Ermittlung des Datenzeitraums

Bevor mit der Ermittlung der Hauptgruppen begonnen werden kann, muss der Zeitraum festgelegt werden, der angibt, welche ausgefüllten App-Fragebögen des App-Anwenders zur Berechnung herangezogen werden. Zu beachten ist, dass eine Lösung gefunden werden muss, die einen optimalen Zeitraum berücksichtigt. Wird für einen App-Anwender eine zu große Datenmenge zur Berechnung der Korrelationskoeffizienten einbezogen (z.B. die letzten zwei Jahre), kann es vorkommen, dass ein App-Anwender einer Gruppe zugewiesen wird, die für ihn "veraltet" und nicht mehr gültig ist. Wenn z.B. vergleichsweise lediglich die Daten des letzten Monats berücksichtigt werden, kann es sein, dass der App-Anwender in einer

anderen Hauptgruppe ist und dadurch aktuelleres Feedback erhält. Aus diesem Grund darf der Zeitraum für die berücksichtigte Datenmenge nicht zu groß sein.

Zunächst wird ermittelt, für welchen Datenzeitraum das Feedback gegeben werden soll. Hierzu werden für den Zeitraum der letzten 30 Tage die Daten der ausgefüllten App-Fragebögen eines App-Anwenders geladen und überprüft, ob innerhalb des Zeitraums mindestens 15 ausgefüllte App-Fragebögen vorhanden sind und die Daten für mindestens einen Einflussfaktor zuverlässig sind (vgl. Schritt 2). Andernfalls wird in 10-Tages-Schritten so lange zurückgerechnet bis der Schwellenwert von 15 ausgefüllten Datensätzen erreicht ist und die Daten zuverlässig sind.

Schritt 2: Berechnung der Datenzuverlässigkeit

Die Datenzuverlässigkeit (Degree of Reliability - DOR) gibt Auskunft darüber, ob innerhalb dieses Zeitraums ausreichend Daten vorhanden sind, um verlässliche Feedbacks geben zu können. Im Folgenden wird erläutert wie der DOR-Wert berechnet wird und welche Bedeutung das Ergebnis hat. Für die Berechnung des DOR-Wertes wird das Verfahren der *Split-half Reliabilität*² verwendet. Im weiteren Verlauf werden die Teilschritte der DOR-Wert-Berechnung genannt:

1. Datenwerte der Tinnitus-Lautheit der Größe nach aufsteigend ordnen.
2. Datenwerte der Einflussfaktoren (Stress, Konzentration, Emotion und Aufregung) richtig sortieren. Die Sortierung richtet sich danach, wie die Werte der Tinnitus-Lautheit sortiert sind. Die Werte der Einflussfaktoren müssen in der selben Reihenfolge, wie der entsprechende Wert der Tinnitus-Lautheit sortiert sein.
3. Geordnete Datenwerte der Tinnitus-Lautheit in zwei Gruppen aufteilen (Lautheit-Gruppe1 und Lautheit-Gruppe2).
4. Datenwerte der Einflussfaktoren in zwei Gruppen aufteilen:
 - Aufteilung der Stress-Werte in zwei Gruppen (Stress-Gruppe1, Stress-Gruppe2)
 - Aufteilung der Konzentrations-Werte in zwei Gruppen (Konzentration-Gruppe1, Konzentration-Gruppe2)
 - Aufteilung der Emotions-Werte in zwei Gruppen (Emotion-Gruppe1, Emotion-Gruppe2)
 - Aufteilung der Aufregungs-Werte in zwei Gruppen (Aufregung-Gruppe1, Aufregung-Gruppe2)
5. Berechnung des Korrelationskoeffizienten, der Tinnitus-Lautheit mit den aufgeteilten Gruppen der entsprechenden Einflussfaktoren:
 - Korrelationskoeffizient1 (Lautheit-Gruppe1 und Stress-Gruppe1)
 - Korrelationskoeffizient2 (Lautheit-Gruppe2 und Stress-Gruppe2)

Nach demselben Prinzip werden für die anderen Einflussfaktoren jeweils zwei Korrelationskoeffizienten berechnet.

6. Zu beiden Korrelationskoeffizienten je Einflussfaktor 1 addieren, um ein positives Ergebnis zu erzielen.

²Auch Testhalbierungsmethode: Aufteilung der Daten in zwei Hälften, an denen anschließend die gleichen Berechnungen vorgenommen werden. Sind die Daten zuverlässig, so sollten die Ergebnisse beider Hälften möglichst gleich sein.

7. Berechnung des DOR-Wertes

Zur Berechnung des DOR-Wertes wird der kleinere Wert der berechneten Korrelationskoeffizienten ermittelt und dieser durch den größeren Wert geteilt. Das Resultat dieser Division ist der DOR-Wert des betrachteten Einflussfaktors. Nach der Durchführung der DOR-Wert-Berechnung existiert insgesamt vier DOR-Werte, für jeden Einflussfaktor einen.

Schritt 3: Prüfung der Datenzuverlässigkeit

Die ermittelten DOR-Werte liegen immer zwischen 0 und 1. In Tabelle 4.3 sind die drei Wertebereiche, die unterschieden werden dargestellt sowie deren Farbkodierung für die Visualisierung in der App (vgl. Kapitel 4.5.2). Bei einem DOR-Wert kleiner als 0.6 sind die Daten nicht verlässlich. Wie am Ende von *Schritt 1* beschrieben ist, wird der Datenzeitraum um weitere zehn Tage erhöht und *Schritt 2* und *Schritt 3* werden erneut ausgeführt. Sind durch dieses Verfahren alle ausgefüllten Fragebögen des App-Anwenders berücksichtigt, ohne dass ein verlässlicher DOR-Wert ermittelt werden kann, wird kein Feedback gegeben, sondern dem App-Anwender angezeigt, dass für diese Kategorie kein Feedback vorhanden ist. Ist der DOR-Wert größer oder gleich 0.6 und kleiner 0.8, sind die Daten einigermaßen verlässlich. Ab einem DOR-Wert von 0.8 gelten die Daten als verlässlich und werden in der App durch die Farbe Grün markiert. Ergibt der DOR-Wert, dass die Daten verlässlich oder einigermaßen verlässlich sind, bedeutet dies, dass ausreichend Daten vorhanden sind und es kann in *Schritt 4* die Berechnung des Korrelationskoeffizienten, für die Zuweisung eines App-Anwenders zu den Hauptgruppen, stattfinden. Werden zu einem Einflussfaktor mindestens einigermaßen verlässliche Daten ermittelt, wird der Anwender darüber informiert, dass wenige Daten für den festgelegten Zeitraum vorhanden sind und dass durch die Betrachtung einer größeren Datenmenge eventuell Feedbacks gegeben werden können.

Tabelle 4.3.: DOR-Wertebereich

Wertebereich	Bedeutung	Farbkodierung
$\text{DOR} < 0.6$	Daten sind nicht verlässlich	Grau (#A7A7A7)
$0.6 \leq \text{DOR} < 0.8$	Daten sind einigermaßen verlässlich	Gelb (#E0F353)
$\text{DOR} \geq 0.8$	Daten sind verlässlich	Grün (#2B9F17)

Schritt 4: Ermittlung der Hauptgruppen

Anhand der ermittelten Daten aus *Schritt 1* bis *Schritt 3*, die durch den DOR-Wert als verlässlich klassifiziert sind, wird der Korrelationskoeffizient für die Einflussfaktoren berechnet, wie in Kapitel 4.2 beschrieben ist. Der App-Anwender wird anschließend den Hauptgruppen zugewiesen, denen die berechneten Korrelationskoeffizienten entsprechen (siehe Tabelle 4.1) und die Ergebnisse werden in der Datenbank gespeichert.

Schritt 5: Feedback-Rückgabe

Der App-Anwender befindet sich in vier der zwölf möglichen Hauptgruppen. Anhand der ermittelten Hauptgruppen, werden dem App-Anwender entsprechende Feedbacks zu jedem der vier Einflussfaktoren gegeben.

Alternative: Zeitraum der Daten anpassen

Alternativ zu *Schritt 1*, der den Datenzeitraum bestimmt für den die Hauptgruppen berechnet

werden sollen, hat der App-Anwender die Möglichkeit diesen Zeitraum individuell zu wählen. Wird dieser Zeitraum an den Hauptgruppen-Algorithmus übergeben, überspringt dieser *Schritt 1* und lädt alle Datensätze des App-Anwenders aus dem angegebenen Zeitraum. Bevor die Hauptgruppen berechnet werden, soll jedoch überprüft werden, ob diese Daten zuverlässig sind. Dies bedeutet der Algorithmus wird nur ausgeführt, wenn mehr als 15 ausgefüllte Fragebögen und verlässliche DOR-Werte vorhanden sind. Jedes Mal, wenn der App-Anwender einen neuen Zeitraum angibt, erfolgt eine neue Berechnung seiner Hauptgruppen und ihm wird eventuell ein anderes Feedback gezeigt. Hierdurch kann beispielsweise verglichen werden, wie sich die Hauptgruppenzuweisung im Laufe der Zeit verändert hat.

4.4.2. Regeldefinition und -auswertung

Als Grundlage zur Regelerstellung für Individuelles-Feedback und Subgruppen-Feedback werden die Fragen aus den verschiedenen Fragebögen und die zugehörigen Antworten der Tinnitus-Betroffenen herangezogen. Administratoren müssen in der Lage sein über eine Benutzeroberfläche auf alle Fragen der Fragebögen zuzugreifen und anhand der möglichen Antwortoptionen Regeln zu definieren. Zu unterscheiden sind einfache und verknüpfte Regeln. Einfache Regeln bestehen aus einer einzigen Bedingung, die zutreffen muss. Verknüpfte Regeln bestehen aus mehreren Bedingungen, die zutreffen müssen, damit ein App-Anwender das Feedback erhält. Weiter können verknüpfte Regeln aus beliebig vielen UND- bzw. ODER-Verknüpfungen bestehen. Bei einer UND-Verknüpfung müssen beide Bedingungen zutreffen, während bei einer ODER-Verknüpfung mindestens eine zutreffen muss [BK106]. Die verschiedenen Möglichkeiten Regeln zu definieren werden nachfolgend als Regeldeklaration in Pseudo-Sprache erläutert.

Beispielregel: Einfache Regel

Tinnitus-Betroffene mit Hörverlust (TSCHQ-Frage 26 zu Hörverlust mit "Ja" beantwortet) erhalten entsprechendes Feedback.

Regel: WENN *Tinnitus-Betroffener einen Hörverlust hat* DANN *gib Feedback 3*.

Beispielregel: UND-Verknüpfung

Tinnitus-Betroffene mit Hörverlust (TSCHQ-Frage 26 zu Hörverlust mit "Ja" beantwortet) und keine Hörgeräte benutzen (TSCHQ-Frage 27 zu Hörgeräte mit "auf keiner Seite" beantwortet) erhalten Feedback, z.B. dass das Tragen von Hörgeräten hilfreich sein kann.

Regel: WENN *Tinnitus-Betroffener einen Hörverlust hat && auf keiner Seite Hörgeräte benutzt* DANN *gib Feedback 4*.

Beispielregel: ODER-Verknüpfung

Tinnitus-Betroffene mit Entspannungsproblemen (Schlimmstes Symptom - Frage 02) oder Tinnitus-Betroffene mit Einschlafproblemen (Mini-TF-Frage 08 mit "stimmt" oder "stimmt teilweise" beantwortet) erhalten entsprechendes Feedback.

Regel: WENN *Tinnitus-Betroffener Entspannungsprobleme hat || Einschlafprobleme hat* DANN *gib Feedback 5*.

Zur Regelauswertung im Feedback-Algorithmus ist die zugeordnete Identifikationsnummer (ID), die zu jeder Frage eines Fragebogens in der Datenbank angegeben ist bedeutsam, denn die Regel, die ein Administrator definiert, soll für jeden App-Anwender überprüft werden. Aus diesem Grund muss zu jeder Frage die gegebene Antwort des App-Anwenders überprüft werden.

Anhand der ID der Frage kann die gegebene Antwort des App-Anwenders aus der Datenbank ermittelt und so überprüft werden, ob die Regel für ihn zutrifft. Das Definieren der Regeln soll für den Administrator so einfach, wie möglich gestaltet werden, indem er sich keine expliziten Gedanken darüber machen muss, welcher Frage welche ID in der Datenbank zugeordnet ist. Um alle Aspekte zur Regeldefinition abzudecken, eignet sich ein Feedback-Formular (vgl. Kapitel 4.5.1), über das er Regeln definieren und hierzu Feedbacks anlegen kann.

4.5. Usability-Konzept

Im weiteren Verlauf werden die Usability-Konzepte für das Feedback-Formular (vgl. Kapitel 4.5.1) und die Android-App (vgl. Kapitel 4.5.2) vorgestellt. Unter dem Begriff Usability ist die Benutzerfreundlichkeit eines Systems zu verstehen. Das System soll dabei einfach erlernbar sein, der Anwender muss es effizient benutzen können, es muss ihm im Gedächtnis bleiben, das System muss eine geringe Fehlerrate aufweisen und soll für den Anwender angenehm zu bedienen sein [Nie10].

4.5.1. Feedback-Formular

Um festzulegen welches Feedback für welche Tinnitus-Gruppe gegeben werden soll, muss ein Konzept entwickelt werden, das es Administratoren erlaubt Feedbacks anzulegen, zu speichern und Regeln zu definieren, die bestimmen, in welchen Fällen das Feedback gegeben wird. Das Konzept für das zu entwickelnde Feedback-Formular orientiert sich an der bestehenden Webseiten-Struktur von TYT. Abbildung 4.4 zeigt die Webseiten-Struktur als Mockup. Bei einem Mockup handelt es sich um erste Entwürfe von Webseiten bzw. Apps vor der Implementierung, um bereits während der Konzeption festzulegen, wie die Inhalte zukünftig strukturiert werden sollen. Untergliedert ist die Webseite in die drei Hauptbereiche Navigationsleiste (1), linke Seitenleiste (2) und Inhaltsbereich (3).

Zur intuitiven Bedienung von Webseiten spielt die Navigation eine wesentliche Rolle, denn über die dadurch gekennzeichneten Pfade gelangen Besucher auf verschiedene Bereiche der Seite. Entscheidend für eine gute Orientierung sind die Art der Navigationsform und die Benennung der Navigationspunkte [Nie10]. Die TYT-Webseite verwendet eine horizontale Navigation. Über die Navigationsleiste (1) findet die Hauptnavigation statt. Angemeldete Benutzer und Administratoren können über eine Seitenleiste weitere Unterpunkte ausführen. Da lediglich Administratoren die Berechtigung für den Zugriff auf das Feedback-Formular haben, betrifft die Erweiterung der Webseite ausschließlich den Navigationspunkt Admin (1). Die linke Seitenleiste in der Administratorenansicht, das sogenannte Admin Panel (2), wird um den Bereich Feedbacks (4) erweitert. Dieser beinhaltet zwei Unterpunkte. Zum einen eine Übersicht ("Overview") aller angelegten Feedbacks, die im Inhaltsbereich (3) dargestellt ist und zum anderen einen Unterpunkt ("Add new"), um neues Feedback anzulegen. Alle vorhandenen Feedbacks sind in Form einer Liste dargestellt (6). Für jedes Feedback ist der Titel ("Feedback Title"), die Hauptgruppenzuweisung ("Maingroup") und eine Kurzbeschreibung ("Short Description") einsehbar. Zusätzlich kann jedes Feedback über einen Button ("Delete") (7) gelöscht werden. Neben der Seitenleiste gelangt ein Administrator auch über einen weiteren Button ("Add feedback")(5) im Inhaltsbereich zum Feedback-Formular.

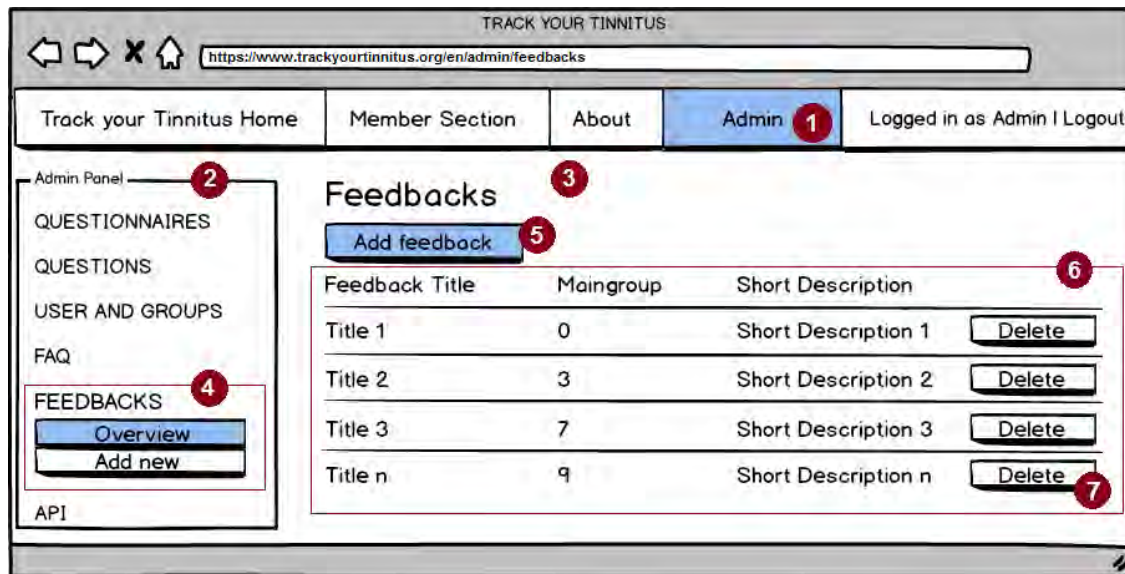


Abbildung 4.4.: Mockup - Feedback-Liste

Der Aufbau des Feedback-Formulars ist in Abbildung 4.5 dargestellt. Es ermöglicht Administratoren Feedbacks anzulegen und zu speichern. Grundlegend ist das Formular durch drei Bereiche aufgebaut, die festlegen, welche Schritte ein Administrator beim Anlegen eines Feedbacks durchführen muss. Die Bereiche des Feedback-Formulars ergeben sich aus den definierten Anwendungsfällen eines Administrators (vgl. Kapitel 3.10). Demnach muss ein Administrator in der Lage sein eine bestimmte Gruppe auszuwählen, für diese optional Regeln zu definieren, die besagen, wann das Feedback gültig ist und für diese Gruppen Feedbacks anzulegen.

Im ersten Schritt (1) bestimmt der Administrator über das Drop-Down-Menü, ob das anzulegende Feedback für eine der definierten Hauptgruppen aus Kapitel 4.2 gültig ist (Hauptgruppen-Feedback), oder ob es für alle Hauptgruppen ("All groups") definiert wird (Allgemeines-Feedback).

Schritt zwei (2) ist optional und bietet die Möglichkeit Regeln, wie in Kapitel 4.4.2 festgelegt zu definieren. Die definierten Regeln legen fest, wann ein Feedback für die im ersten Schritt definierte Gruppe gültig ist. Sind im ersten Schritt alle Gruppen ausgewählt und wird im zweiten Schritt eine Regel angegeben, kann Individuelles-Feedback erstellt werden. Ist im ersten Schritt eine Hauptgruppe ausgewählt und im zweiten Schritt zusätzlich eine Regel definiert, handelt es sich um ein Subgruppen-Feedback. Über das Dropdown-Menü „Select a question“ lässt sich eine der Fragen aus den statistischen Fragebögen auswählen. Zur gewählten Frage werden die Antwortmöglichkeiten angegeben und es muss diejenige Antwort ausgewählt werden, für die die Regel zutreffen soll. Durch den Button „Add Rule“ kann die Regel hinzugefügt und im Eingabefeld ("feedback rule") angezeigt werden. Eine Regel kann durch die Buttons zu einer Und- ("AND") bzw. eine Oder-Verknüpfung ("OR") ergänzt werden.

Der dritte Schritt (3) sieht vor, dass die Feedback-Details vervollständigt werden. Im oberen Teil lassen sich die Inhalte in Englisch erstellen, darunter in Deutsch. Durch den Ansatz, Inhalte

bei Bedarf anzuzeigen, indem Punkte aufgeklappt werden, ermöglicht es zu einem späteren Zeitpunkt beliebige Sprachen hinzuzufügen und dennoch die Übersichtlichkeit des Formulars zu gewährleisten. Zu einem Feedback zählen der Titel ("Feedback Title"), eine Kurzbeschreibung ("Short Description"), eine detaillierte Feedback-Beschreibung ("Feedback Description") sowie ein Eingabefeld zur Literaturangabe.

Im vierten Schritt (4) wird das Feedback-Formular gespeichert und alle Informationen werden in entsprechenden Datenbanktabellen gesichert (vgl. Kapitel 5.2). Der Aufbau und das Design des Feedback-Formulars orientieren sich an der bestehenden TYT-Webseite und den bestehenden Formularen.

Track your Tinnitus Home Member Section About **Admin** Logged in as Admin | Logout

Admin Panel

QUESTIONNAIRES

QUESTIONS

USER AND GROUPS

FAQ

FEEDBACKS

Overview

Add new

API

Add Feedback for Maingroups and Tip of the Week

- 1. Choose Maingroup**
Choose "None" if you want to add a general Feedback, that only applies to Tip of the Week.
All groups
Maingroup 1
Maingroup 2
Maingroup 3
...
- 2. Define Rule**
feedback rule
Select a question
Question 1
Question 2
Question 3
Question 4
...
Add rule AND OR
- 3. Complete Feedbackdetails**
v Englisch
Feedback Title
Short Description
Feedback Description
> German
- 4** **Save**

Abbildung 4.5.: Mockup - Feedback-Formular

4.5.2. Android-App

Abbildung 4.6 veranschaulicht das Navigationskonzept aus Sicht des App-Anwenders. Über das Hauptmenü (1) kann der App-Anwender zusätzlich zur vorherigen Version einsehen, mit welchen Einflussfaktoren sein Tinnitus zusammenhängt ("My influencing factors")(2). Weiter kann er den Tipp der Woche ("Tip of the week") einsehen (4) und selbst ein Feedback geben ("App assessment")(7).

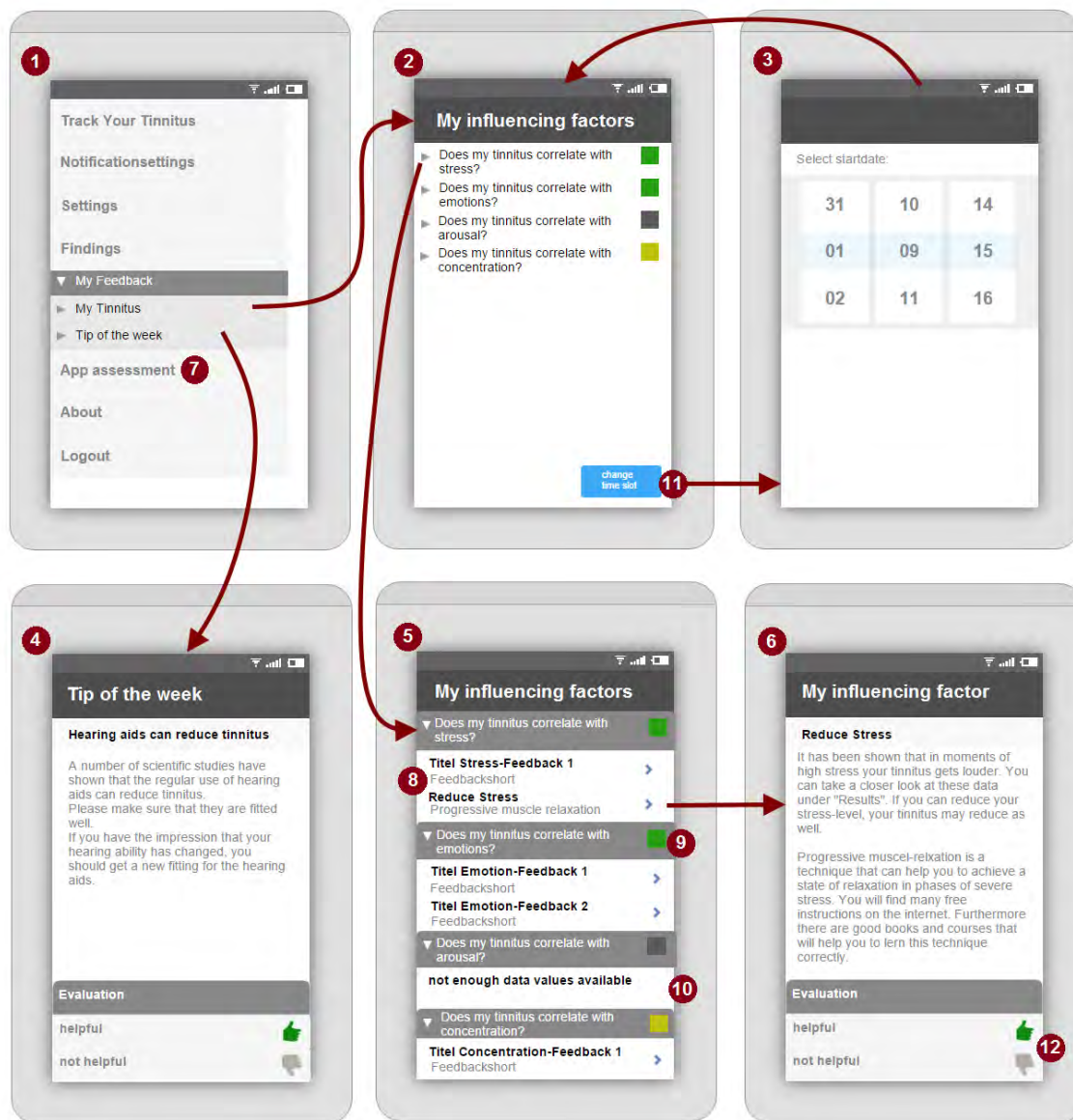


Abbildung 4.6.: App-Mockups mit Navigationskonzept

Zu den individuellen Einflussfaktoren erhält der App-Anwender, sofern er ausreichend oft den Fragebogen in der App ausgefüllt hat, in der Einflussfaktoren-Ansicht (5) je Einflussfaktor eine

Liste mit Feedbacks (8). Bei diesen Feedbacks handelt es sich um Hauptgruppen-Feedbacks. Die farblich gekennzeichneten Quadrate (9) symbolisieren die Datenzuverlässigkeit. Auf Grundlage des Zeitraums, in dem die Daten betrachtet werden steht die Farbe grün dafür, dass die Daten zum jeweiligen Einflussfaktor eine zuverlässige Aussage treffen. Gelb bedeutet, dass die Daten einigermaßen zuverlässig sind und grau deutet auf unzuverlässige Daten hin. In diesem Fall erhält der App-Anwender kein Feedback, sondern einen Hinweis (10), dass nicht ausreichend Daten vorhanden sind, um ein zuverlässiges Feedback zu geben. Er erhält zusätzlich den Hinweis, dass er das Zeitfenster für den Berechnungszeitraum größer machen kann und ferner, dass es vorkommen kann, dass das Ergebnis immer eine unzuverlässige Datenmenge ergibt. Das Zeitfenster kann der App-Anwender unabhängig des gegebenen Feedbacks (2) jederzeit über den Button ("change time slot") am unteren Bildschirmrand ändern (11). Dadurch kann er durch die Angabe eines Start- (3) und Enddatums angeben, ob er beispielsweise die Daten der letzten zwei Monate, der letzten fünf Monate oder des letzten Jahres berücksichtigen möchte. Je nach Zeitangabe können sich die Feedback-Vorschläge ändern. Wählt er ein Feedback aus der Liste aus, erhält er eine Detailbeschreibung (6). Das erhaltene Feedback kann der App-Anwender selbst bewerten, ob es für ihn hilfreich oder nicht hilfreich ist (12). Jede Woche erhält er einen neuen Tipp der Woche (4), den er auch bewerten kann. Bei den Feedbacks zum Tipp der Woche kann es sich um Allgemeines-Feedback, Individuelles-Feedback oder Subgruppen-Feedback handeln.

Das Design der App orientiert sich am bestehenden Design der Android-App und ergänzt dieses um zusätzliche Farben für die grafische Darstellung der Datenzuverlässigkeit. Tabelle 4.3 listet die entsprechenden Farbwerte auf.

5

Architektur und Implementierung

Dieses Kapitel beschreibt die Umsetzung des in Kapitel 4 vorgestellten Konzepts. In Kapitel 5.1 werden die Komponenten, um die die TYT-Architektur erweitert wird in Form einer Gesamtübersicht erläutert. Kapitel 5.2 beschreibt, um welche Datenbanktabellen die Datenbank erweitert wird und wie diese Tabellen aufgebaut sind. Kapitel 5.3 stellt das realisierte Feedback-Formular zum Anlegen von Feedbacks und Regeln für Administratoren vor. Die Implementierung der Android-App wird in Kapitel 5.4 erklärt. Kapitel 5.5 erläutert die Kommunikation und den Datenaustausch zwischen der App und dem Feedback-Algorithmus. Dessen detaillierte Implementierung ist in Kapitel 5.6 erläutert.

5.1. Gesamtübersicht

Zur Implementierung dieser Arbeit werden drei Komponenten entwickelt: das Feedback-Formular, der Hauptgruppen-Algorithmus und die Regelauswertung. Abbildung 5.1 zeigt, wie diese Komponenten in die TYT-Architektur integriert sind. Die Architektur wird dabei in vier Schichten betrachtet.

Die Android-App stellt die Präsentationsschicht dar. Der App-Anwender kann dort Feedbacks, wie in Kapitel 4.5.2 beschrieben einsehen. Dazu wird der bestehende REST-Client der Android-App verwendet, um passende Feedbacks von der Datenbank abzurufen. Die TYT-Web-Anwendung stellt eine REST-API bereit, die um die entsprechenden Schnittstellen erweitert wird (vgl. Kapitel 5.5) und dem REST-Client die Feedbacks in einem vordefinierten Format liefert.

In der Interaktionsschicht wird die TYT-Web-Anwendung um die Komponente *Feedback-Formular* erweitert. Das Feedback-Formular wird basierend auf dem Konzept aus Kapitel 4.5.1 entwickelt und ermöglicht dem Administrator das Verwalten von Feedbacks. Die Komponente speichert und lädt Feedbacks aus der MySQL-Datenbank und ist in Kapitel 5.3 beschrieben.

Hauptbestandteil der Arbeit ist der *Hauptgruppen-Algorithmus*. Dieser ist zusammen mit der *Regelauswertung* in der Logikschicht eingeordnet. Die beiden Komponenten enthalten Funktionen zur Berechnung der Hauptgruppen, zur Auswertung von Regeln sowie zur Ermittlung geeigneter Feedbacks. Sie werden von der REST-API aufgerufen.

In der Datenhaltungsschicht ist die MySQL-Datenbank eingeordnet, die für die Speicherung aller Anwendungsdaten zuständig ist. Die Datenbank wird um Tabellen zur Verwaltung von Hauptgruppen und Feedbacks erweitert.

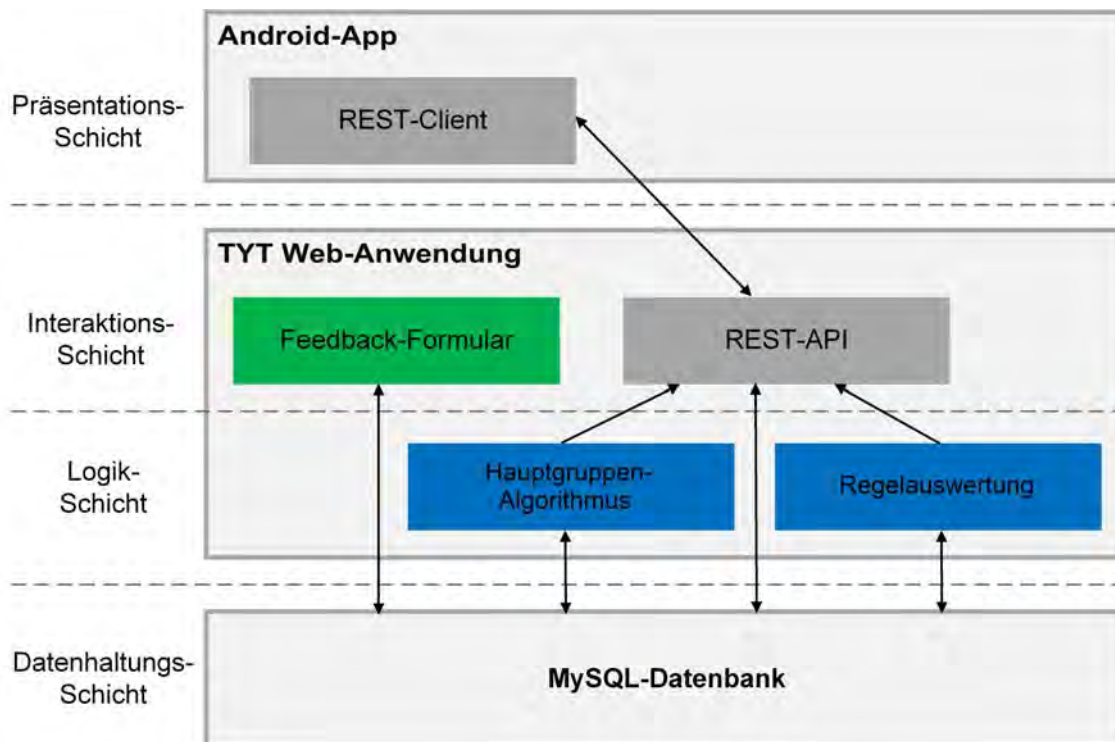


Abbildung 5.1.: Architektur-Gesamtübersicht

5.2. Datenstrukturen

Die Speicherung der Daten erfolgt über eine MySQL-Datenbank (vgl. Kapitel 5.1). Diese Datenbank ist im TYT-Projekt bereits vorhanden und stellt eine Reihe von Tabellen zur Verfügung. Für diese Arbeit sind die bestehenden Tabellen „questions“, „standardanwers“ und „answers“ von Bedeutung. Diese enthalten die Daten, auf die der Feedback-Algorithmus zugreift (vgl. Kapitel 5.6). Tabelle „questions“ enthält die Fragen aller Fragebögen (vgl. Tabelle A.2 - A.4). Tabelle „standardanwers“ speichert alle Antworten der App-Anwender zum App-Fragenbogen (vgl. Tabelle A.1) und nimmt die Schwankungen der Tinnitus-Wahrnehmung in verschiedenen Bereichen auf. Die Antworten der App-Anwender zu den statistischen Fragebögen, sind in der Tabelle „anwers“ gespeichert.

Da diese Arbeit die Funktionalität der *Mobile Crowd Sensing Plattform* TrackYourTinnitus um einen Feedback-Algorithmus erweitert, muss die Datenstruktur erweitert werden. Es werden zum einen Tabellen für die Hauptgruppenzuordnung und zum anderen Tabellen zur Feedbackverwaltung benötigt.

Datenstruktur Feedbacks

Um einzelne funktionale Anforderungen aus Kapitel 3.4.1 zu erfüllen, werden folgende drei Tabellen benötigt: „feedbacks“, „feedbacklocalizations“ und „feedbackevaluations“. Die Struktur dieser Tabellen und ihr Zusammenhang sind in Abbildung 5.2 dargestellt.

Die Tabelle „feedbacks“ enthält alle angelegten Feedbacks, die über das Feedback-Formular (vgl. Kapitel 5.3) vom Administrator angelegt werden. Alle Feedbacks, die in dieser Tabelle

gespeichert sind, werden vom Hauptgruppe-Algorithmus überprüft. Jedes Feedback besteht aus einer Identifikationsnummer ("id"), einer Hauptgruppe ("maingroup"), wobei hier auch die Option "Alle Hauptgruppen" zulässig ist, einem Titel ("feedbacktitle"), einer Kurzbeschreibung ("feedbackshort") und einer Feedback-Beschreibung ("feedbackdescription"). Die Felder "rule" und "fields" dienen zur Angabe und Auswertung von Regeln und können auch leer sein, sofern für ein Feedback keine Regel definiert ist. Das Literaturfeld ("literature") ist ebenfalls optional und dient dem Administrator zum Speichern von Literaturquellen zu den Feedbacks. Die Inhalte der Feedback-Tabelle werden in Englisch angelegt.

Zur Unterstützung zukünftiger Projekte, die sich mit dem Thema Mehrsprachigkeit befassen, werden deutsche Inhalte in der Tabelle „feedbacklocalizations“ gespeichert. Weitere Sprachen können ebenfalls in dieser Tabelle gespeichert werden. Die Verbindung der beiden Tabellen erfolgt durch den Primärschlüssel ("id"), der Feedback-Tabelle und dem Fremdschlüssel ("feedback_id") in "feedbacklocalization". Neben den deutschsprachigen Inhalten enthält die Tabelle außerdem einen Sprachcode ("languagecode"), der die Sprache definiert, z.B. *DE* für Deutsch.

In der Tabelle „feedbackevaluations“ werden die Feedback-Bewertungen der App-Anwender gespeichert. Wie die Tabelle „feedbacklocalizations“, werden auch in dieser Tabelle die Datensätze über den Fremdschlüssel ("feedback_id") den Feedbacks zugeordnet. Die Tabelle enthält außerdem ein Feld "user_id", zur Identifikation des App-Anwenders, der die Feedback-Bewertung gegeben hat und ein Feld "helpful", das anzeigt, ob das Feedback hilfreich oder nicht hilfreich für diesen App-Anwender ist.



Abbildung 5.2.: Datenstruktur - Feedbacks

Datenstruktur Hauptgruppen

Die folgenden Tabellen dienen dem Feedback-Algorithmus, um die Werte der verschiedenen App-Anwender zu speichern und ihnen dadurch geeignete Feedbacks zu geben. Die Struktur der Tabellen ist in Abbildung 5.3 dargestellt.

Die Hauptgruppen-Tabelle ("maingroups") enthält die definierten Hauptgruppen (vgl. Tabelle 4.1), denen die App-Anwender zugewiesen werden können. Diese sind fest in der Datenbank gespeichert und können jederzeit erweitert werden. Eine Hauptgruppe ist durch eine ID, einen Namen ("name") und eine Beschreibung ("description") definiert.

Da die übrigen drei Tabellen "maingroupuservalues", "userfeedbackgroups" und "usermemories" die Informationen eines App-Anwenders erweitern, sind sie über den Fremdschlüssel "user_id" mit der bestehenden Benutzertabelle ("users") der TYT-Plattform verbunden.

In der Tabelle "maingroupuservalues" werden die aktuellen Korrelationskoeffizienten und DOR-Werte jedes App-Anwenders gespeichert, die durch den Hauptgruppen-Algorithmus (vgl. Kapitel 5.6.1) berechnet werden. Dabei enthält die Tabelle für jeden App-Anwender einen Datensatz, der bei wiederholter Ausführung des Hauptgruppen-Algorithmus überschrieben wird.

Zusätzlich zu den Korrelationskoeffizienten und DOR-Werten speichert der Hauptgruppen-Algorithmus in der Tabelle "userfeedbackgroups" für jeden App-Anwender die vier Hauptgruppen, denen er zugewiesen ist.

Die Tabelle "usermemories" dient, um zusätzliche Werte für App-Anwender zu speichern. Beispielsweise werden für jeden App-Anwender im Feld "used_tipoftheweeks" alle bereits gegebenen Feedbacks für den Tipp der Woche über die ID ("feedback_id") gespeichert (vgl. Kapitel 5.5.2).

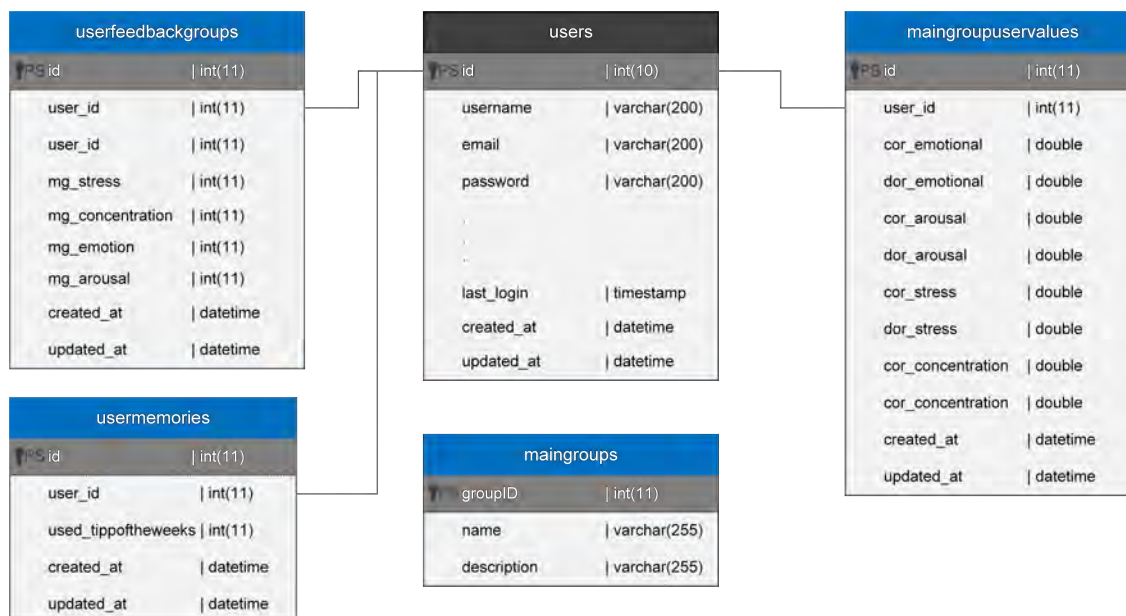


Abbildung 5.3.: Datenstruktur - Hauptgruppen

5.3. Feedback-Formular

Zum Anlegen von Feedbacks in der TYT-Web-Anwendung wird eine Seite zur Übersicht der Feedbacks, eine Seite zur Eingabe der Feedbacks und ein Feedback-Controller benötigt. Das entwickelte Feedback-Formular zur Eingabe von Feedbacks ist in Abbildung 5.4 dargestellt und entspricht dem Mockup aus Kapitel 4.5.1.

Add Feedback for Maingroups and Tip of the Week

1. Choose Maingroup

Choose "All Groups" if you want to add a general Feedback, that only applies to Tip of the Week. Maingroups are based on AppData.

2. Define Rule (optional)

First select a question, to add rules. To connect rules use the AND/OR-Button and please consider the brackets (only use round brackets).

Narrows the Feedback to users, that gave answers in a specific way. If no rule is given, the Feedback will apply to all users in the selected Maingroup.

Rules are based on the registration questionnaires.

[> Example](#)

3. Complete Feedback Details

4. Note Feedback literature

Abbildung 5.4.: Feedback-Formular

Der Abruf von Hauptgruppen für die Darstellung im Dropdown-Menü, sowie der Abruf von Fragen aus der Datenbank zum Definieren von Regeln und das Speichern von Feedbacks in die Datenbank, wird durch *Eloquent ORM* (Object-Relationship-Mapping) von Laravel [Otw15] realisiert. Dieses ermöglicht Datensätze aus Datenbanktabellen in PHP [PHP15] als Objekte zu laden und diese bei der Programmierung zu verwenden. Listing 5.1 veranschaulicht, wie aus der Datenbanktabelle "maingroups" (vgl. Kapitel 5.2) die definierten Hauptgruppen geladen werden (Zeile 3). Um sie dem Administrator zur Auswahl im Dropdown-Menü anzuzeigen, werden die Hauptgruppen im Feedback-Controller in ein Array geschrieben (Zeile 4). Das Array wird in der Variable *\$maingroups* an das Feedback-Formular übergeben. Dort wird es zur Darstellung der Hauptgruppen im Dropdown-Menü als Quelle verwendet, wie Listing 5.2 (Zeile 3) zeigt.

Listing 5.1: Datenbankaufruf von Hauptgruppen

```

1 // Get all Maingroups from DB and write them in an Array
2 $i = 0;
3 foreach (Maingroup::all() as $value) {
4     $maingroups[$i] = $value->name;
5     $i = $i + 1;
6 }

```

Listing 5.2: Hauptgruppenanzeige in Feedback-Formular

```

1 <div class="form-group">
2     {{ Form::label('maingroup', 'Choose Maingroup') }}
3     {{ Form::select('maingroup', $maingroups) }}
4 </div>

```

Abbildung 5.5 veranschaulicht, wie über das Feedback-Formular eine Regel erstellt werden kann. Zunächst wird eine Frage mit ID ("40. Does your tinnitus more sound like a tone or more like noise") selektiert und zu dieser Frage die Antwort ('tone') als Bedingung definiert, die zutreffen muss, damit ein App-Anwender das Feedback erhält. Durch die Buttons "AND" und "OR" lassen sich Verknüpfungen erstellen und durch "Add rule" wird diese Regel zur bestehenden Regel hinzugefügt.

2. Define Rule (optional)

First select a question, to add rules. To connect rules use the AND/OR-Button and please consider the brackets (only use round brackets).

Narrows the Feedback to users, that gave answers in a specific way. If no rule is given, the Feedback will apply to all users in the selected Maingroup. Rules are based on the registration questionnaires.

> Example

((30 == 'true') || (28 == 'true')) && (40 == 'tone ')

40. Does your tinnitus more sound like a tone or more like noise ?

'tone '
 'noise '
 'crickets '
 'other'

Add Rule
AND
OR

Abbildung 5.5.: Beispiel - Regeldefinition

Abbildung 5.6 zeigt zwei Optionen, wie Regeln definiert werden können. Beide Optionen bestehen hierbei aus den selben Regeln, haben aber eine unterschiedliche Und- bzw. Oder-Verknüpfung. Die Regeln bestehen aus drei Vergleichen. Jeder Vergleich besteht aus einer ID der Frage, einem Vergleichsoperator und der Konstante mit der die Frage in der Regelauswertung verglichen wird (vgl. Kapitel 5.6.2). Option 1 prüft ob der Tinnitus-Betroffene Einschlaf- oder Entspannungsprobleme hat und ob er den Tinnitus als Ton wahrnimmt. Dabei muss nur einer der ersten beiden Vergleiche wahr sein. Option 2 prüft zunächst ob der Tinnitus-Betroffene Einschlafprobleme hat oder ob er Entspannungsprobleme hat und den

Tinnitus als Ton wahrnimmt. Hierbei müssen sowohl Vergleich 2 als auch Vergleich 3 zutreffen. Bei der Regelerstellung im Feedback-Formular muss daher beachtet werden, dass die runden Klammer richtig gesetzt sind.

Option 1:
`((30 == 'true') || (28 == 'true')) && (40 == 'tone')`

Option 2:
`(30 == 'true') || ((28 == 'true') && (40 == 'tone'))`

Vergleich 1
Vergleich 2
Vergleich 3

Abbildung 5.6.: Beispiel - Optionen zur Regeldefinition

Das im Feedback-Formular erstellte Feedback, wird in PHP als Objekt angelegt und bearbeitet, wie in Listing 5.3 (Zeile 2) veranschaulicht ist. Zum Feedback-Objekt ist eine Eloquent-Klasse definiert, die ihre Struktur aus der Datenbanktabelle bezieht. Ist das Feedback-Formular vollständig ausgefüllt, werden die Parameter des Feedback-Objekts gesetzt, indem die Eingabe des Administrators durch den Funktionsaufruf `Input::get('parametername')` abgerufen wird. Anschließend wird das Feedback-Objekt mit Hilfe von Eloquent ORM durch die Bereitstellung der `save`-Methode in der Datenbank gespeichert (Zeile 12).

Listing 5.3: Feedback in Datenbank speichern

```

1 // Create new Feedback-Object
2 $feedback = new Feedback();
3 // Fill Feedback
4 $feedback->feedbacktitle = Input::get('feedbacktitle');
5 $feedback->maingroup = Input::get('maingroup');
6 $feedback->feedbackdescription = Input::get('feedbackdescription');
7 $feedback->feedbackshort = Input::get('feedbackshort');
8 $feedback->rule = Input::get('rule');
9 $feedback->fields = Input::get('fields');
10 $feedback->literature = Input::get('literature');
11 // Save Feedback to Database
12 $feedback->save();

```

5.4. Android-Applikation

Im Folgenden wird beschrieben, wie das in Kapitel 4.5.2 vorgestellte Konzept der Android-App implementiert ist. Dabei werden die drei Aktivitäten *Einflussfaktoren*, *Tipp der Woche* und *Feedback geben* sowie die Funktion *Feedback bewerten* betrachtet. Diese sind anhand der in Kapitel 3.4 definierten Anforderungen entwickelt. Abbildung 5.7 zeigt das Hauptmenü der App mit den drei zusätzlichen Aktivitäten.

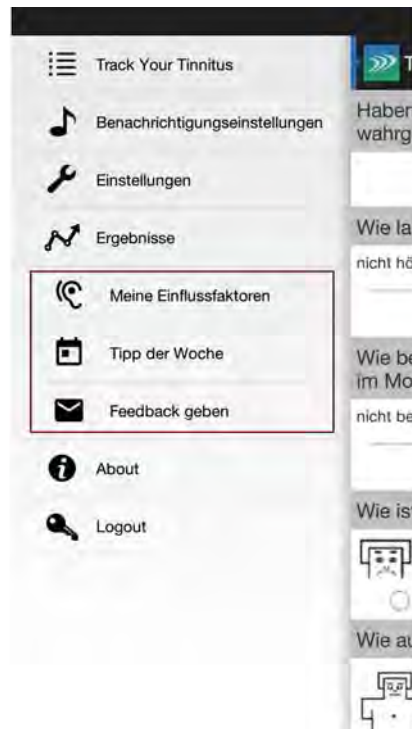


Abbildung 5.7.: Screenshot - Hauptmenü Android-App

5.4.1. Einflussfaktoren

Beim Ablauf zur Ermittlung, ob und welche Feedbacks in der Einflussfaktoren-Ansicht angezeigt werden können, ist zu unterscheiden, ob Internet vorhanden ist, Feedbacks zu einem früheren Zeitpunkt bereits geladen wurden und ob der App-Anwender einen Datumsbereich angegeben hat, für den die Feedbacks geladen werden sollen. Abbildung 5.8 zeigt diesen Ablauf.

Zunächst wird überprüft, ob Internet vorhanden ist und sofern dies der Fall ist folgt eine weitere Prüfung dahingehend, ob ein Datumsbereich mit Start- und Enddatum durch den App-Anwender ausgewählt ist. Trifft dies zu, dient der Datumsbereich als Zeitintervall für das Feedback, das von der Datenbank geladen werden soll. Das Laden der Feedbacks erfolgt über eine entsprechende Schnittstelle (vgl. Kapitel 5.5.1), die den Hauptgruppen-Algorithmus (vgl. Kapitel 5.6.1) mit dem Zeitintervall aufruft und die ermittelten Feedbacks für diesen Zeitraum zurück gibt. Diese Feedbacks werden durch die App gespeichert und in der Einflussfaktoren-Ansicht angezeigt.

Ist Internet vorhanden, aber kein Datumsbereich ausgewählt, beispielsweise beim ersten Ausführen der Einflussfaktoren-Ansicht, erfolgt über dieselbe Schnittstelle der Aufruf des Hauptgruppen-Algorithmus, jedoch mit dem Unterschied, dass dieser selbst einen Datumsbereich berechnen muss, für den das Feedback gegeben werden soll (vgl. Kapitel 5.6.1). Die geladenen Feedbacks werden gespeichert und entsprechend angezeigt. Für den Fall, dass kein Internet vorhanden ist, aber zu einem früheren Zeitpunkt ein Feedback-Abruf stattgefunden hat, werden die zuletzt gespeicherten Feedbacks angezeigt. Ist kein Internet

vorhanden und es ist noch nie ein Feedback geladen worden, erhält der App-Anwender einen Hinweis, dass Internet erforderlich ist.

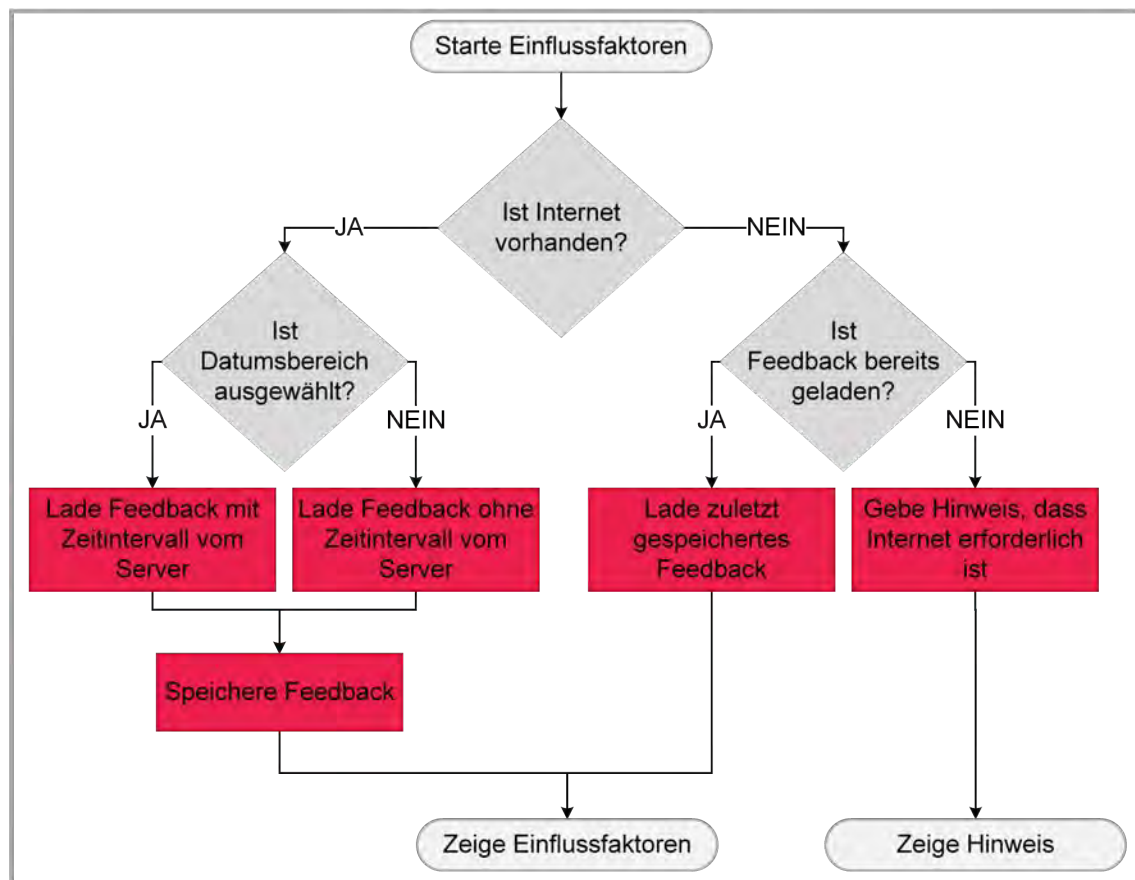


Abbildung 5.8.: Ablaufdiagramm - Einflussfaktoren

Um die ermittelten Feedbacks in der TYT-Android-App anzuzeigen werden zwei Ansichten benötigt: *MaingroupListFragment*, um die Hauptgruppen-Feedbacks der Einflussfaktoren als Liste anzuzeigen und *MaingroupDetailActivity*, um ein ausgewähltes Hauptgruppen-Feedback mit allen Details anzuzeigen. Diese bestehen jeweils aus einer Java-Klasse und einer XML-Datei, die das Layout der Ansicht bestimmt und die anzuzeigenden Elemente enthält. Abbildung 5.9 zeigt die Ansicht der aufgelisteten Einflussfaktoren. Die Feedbacks sind in der Liste in die vier Kategorien der Einflussfaktoren gegliedert und sind dabei ein- und ausklappbar. Die Ansicht enthält außerdem am unteren Bildschirmrand eine Erklärung zur Verlässlichkeit der Einflussfaktoren, indem die Bedeutung der Farben erklärt wird. Zudem wird die Bedeutung der Farben auch erläutert, wenn eine Farbe in einem Einflussfaktor ausgewählt wird. Darüber hinaus wird über einen Info-Button näher beschrieben, worum es sich bei dem Begriff der Datenzuverlässigkeit handelt. Wird in der Liste ein Feedback ausgewählt, wird die dritte in Abbildung 5.9 dargestellte Ansicht geladen. Hier werden der Titel und die Beschreibung des ausgewählten Feedbacks angezeigt und es kann eine Bewertung des Feedbacks erfolgen (vgl. Kapitel 5.4.3).

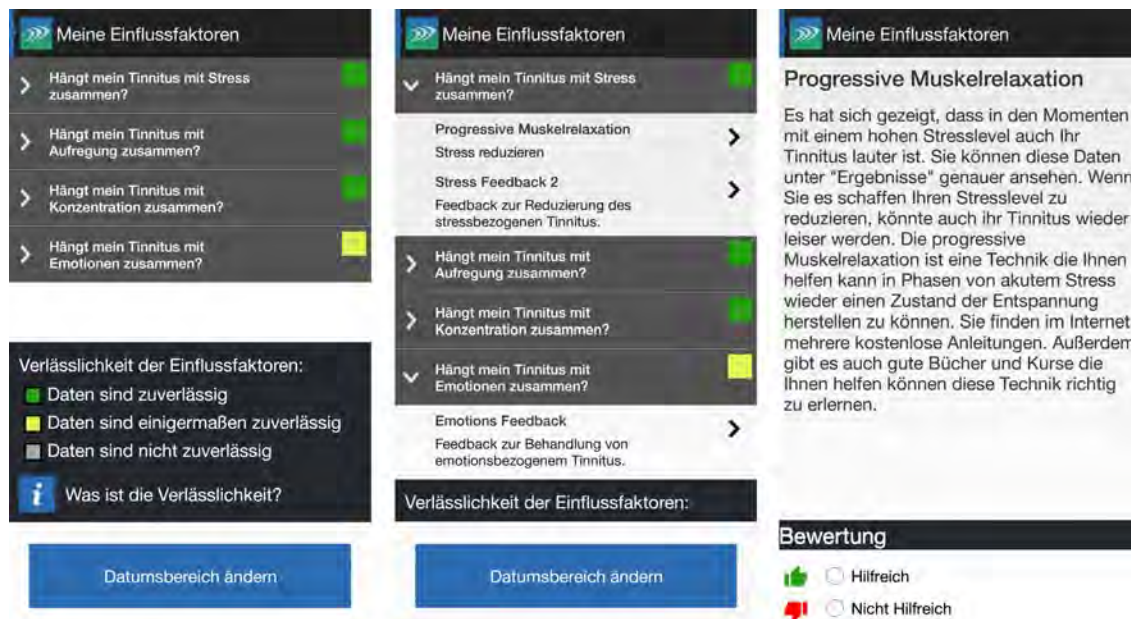


Abbildung 5.9.: Screenshot - Einflussfaktoren und Feedback-Ansicht

Die Ansicht *MaingroupListFragment* erweitert die Klasse *SherlockFragment* [Wha12] und stellt eine *ExpandableListView* zur Verfügung, in der die Hauptgruppen-Feedbacks sortiert eingefügt werden können [Goo15]. Hierzu ist ein *ExpandableListAdapter* implementiert, der für jeden Einflussfaktor eine Liste von Feedbacks in Form einer *LinkedHashMap*¹, entgegennimmt und diese anzeigt [Bac12]. Feedbacks werden dabei als Java-Objekte von der Klasse *Feedback* angelegt.

Um die Feedbacks abzurufen, werden diese über die Schnittstelle der TYT-Web-Anwendung (vgl. Kapitel 5.5) geladen. Hierzu wird die Methode *loadFeedback* (vgl. Listing 5.4) aufgerufen, die die Anmeldeparameter setzt (Zeile 3-4) und über die Klasse *FeedbackDownload* das Herunterladen der Feedbacks einleitet. Die Klasse *FeedbackDownload* ruft die REST-Schnittstelle der TYT-Web-Anwendung auf und übergibt der Schnittstelle die Parameter mit den Anmeldedaten (Zeile 17).

Ist der REST-Aufruf erfolgreich, erhält die Callback-Methode die Feedbacks der Einflussfaktoren als JSON-Objekt, in demjenigen Format, das durch die Schnittstellenbeschreibung (Kapitel 5.5) definiert ist. Die Callback-Methode ruft anschließend die Methode *finishedLoading*, der Aktivität *MaingroupListFragment* auf und übergibt dieser Methode das JSON-Objekt.

Listing 5.4: Feedback-Download

```

1 private void loadFeedback(String startDate, String endDate) {
2     ...
3     RequestParams params = new RequestParams();
4     params.put("access_token", accessToken);
5     ...

```

¹Bei einer *LinkedHashMap* handelt es sich um eine Java-Klasse, die eine geordnete Liste von Schlüssel-Wert-Paaren enthält.

```

6 FeedbackDownload asynAnswers = new FeedbackDownload(this);
7 asynAnswers.execute(new RequestParams[] { params });
8 }
9 ...
10
11 private class FeedbackDownload extends AsyncTask<RequestParams,
12     Void, Boolean> {
13     ...
14     @Override
15     protected Boolean doInBackground(RequestParams... params) {
16         RequestParams requestParams = params[0];
17         RestClient.get("api/maingroups", requestParams,
18             new CustomJsonHandler(mActivity) {
19                 @Override
20                 public void onSuccess(JSONObject response) {
21                     MaingroupListFragment activity = (MaingroupListFragment)
22                         this.mActivity;
23                     activity.finishedLoading(response);
24                 }
25                 ...
26                 return null;
27             }
28         }
29     }

```

In der Methode *finishedLoading* wird das JSON-Objekt "geparst", d.h. es werden die Feedbacks und die DOR-Werte, die das Objekt enthält, entnommen und in die Listen der *ExpandableListView* geschrieben. Listing 5.5 zeigt hierzu die Listen-Objekte (Zeile 2-4), die benötigt werden um die Feedbacks anzuzeigen und wie diese im *ExpandableListAdapter* gesetzt werden müssen (Zeile 9-12). Sind die Listen gefüllt, zeigt die *ExpandableListView* diese an. Der letzte Schritt der Methode *finishedLoading* ist es die Listen in den *SharedPreferences* [Bac12] zu speichern (Zeile 17-30), damit diese auch dann angezeigt werden können, falls der App-Anwender zu einem späteren Zeitpunkt über kein Internet verfügt.

Listing 5.5: Feedback-Anzeige

```

1 public void finishedLoading(JSONObject maingroup_json){
2     expandableListTitle = new ArrayList<String>();
3     expandableListDetail = new LinkedHashMap<String,
4         ArrayList<Feedback>>();
5
6     // Parse JSON-Object and put them in the Lists for the
7     //expandableListView
8     ...
9     // Set the List Adapter for the expandableListView
10    final ExpandableListAdapter expandableListAdapter =
11        new ExpandableListAdapter(getActivity(), expandableListTitle,
12            expandableListDetail, dorValues);

```

```

13  ...
14  expandableView.setAdapter(expandableListAdapter);
15  ...
16  // Save Feedbacklists in Prefs for future use
17  SharedPreferences prefs =
18      getActivity().getSharedPreferences(PREFS_NAME, 0);
19  SharedPreferences.Editor editor = prefs.edit();
20  try {
21      editor.putString("dorValues",
22          ObjectSerializer.serialize(dorValues));
23      editor.putString("expandableListTitle",
24          ObjectSerializer.serialize(expandableListTitle));
25      editor.putString("expandableListDetail",
26          ObjectSerializer.serialize(expandableListDetail));
27  } catch (IOException e) {
28      e.printStackTrace();
29  }
30  editor.commit();
31  }

```

Jedes Feedback kann über einen *OnChildClickListener* angeklickt werden. Dadurch wechselt die App in die Ansicht *MaingroupDetailActivity*. Listing 5.6 zeigt diesen Klick-Listener, der zunächst das geklickte Feedback ermittelt (Zeile 3) und anschließend eine neue Aktivität anlegt und startet (Zeile 7). Diese Aktivität ist vom Typ *MaingroupDetailActivity* und erhält das ausgewählte Feedback als Übergabe (Zeile 10).

Listing 5.6: Klick-Listener für Listenelemente

```

1  public boolean onChildClick(ExpandableListView parent, View v,
2      int groupPosition, int childPosition, long id) {
3      Feedback f = (Feedback)expandableListDetail.get(
4          expandableListTitle.get(groupPosition)).get(childPosition);
5      if (f.id != 0){
6          Intent i;
7          i = new Intent(getActivity().getBaseContext(),
8              MaingroupDetailActivity.class);
9          // set Feedback as parameter for the activity
10         i.putExtra("feedback", f);
11         startActivity(i);
12         return false;
13     }
14     else{
15         return false;
16     }
17 }
18 }

```

Innerhalb der Ansicht *MaingroupDetailActivity* wird das zuvor gewählte Hauptgruppen-Feedback angezeigt. Dazu wird es, wie in Listing 5.7 zu sehen ist geladen (Zeile 3) und die Titel- und Beschreibungs-Texte werden aus dem Feedback-Objekt gesetzt (Zeile 11-13).

Listing 5.7: Anzeige Feedback-Details

```

1 // Get Feedback from Intent
2 Intent i = getIntent();
3 feedback = (Feedback)i.getSerializableExtra("feedback");
4
5 // Get Views
6 TextView title = (TextView) findViewById(R.id.feedbacktitle);
7 TextView description =
8     (TextView) findViewById(R.id.feedbackdescription);
9
10 // Set Texts
11 setTitle(R.string.maingroupTitle);
12 title.setText(feedback.name);
13 description.setText(feedback.description);

```

5.4.2. Tipp der Woche

Die Ansicht des Tipp der Woche ist ähnlich der Detailansicht der Einflussfaktoren und wird durch die Java-Klasse und das Layout-XML *TippoftheWeekFragment* bestimmt (vgl. Abbildung 5.10).

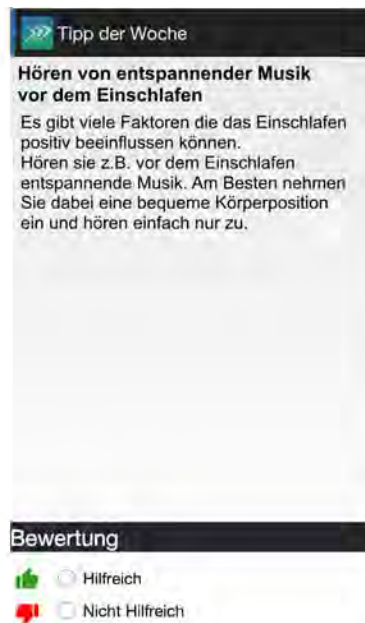


Abbildung 5.10.: Screenshot - Tipp der Woche

Wird die Ansicht Tipp der Woche ausgewählt, erfolgt innerhalb der App eine Überprüfung, ob bereits ein Tipp der Woche vorhanden ist (vgl. Abbildung 5.11). Ist keiner vorhanden, bedeutet dies, dass diese Ansicht zum ersten Mal abgerufen wird. Weiter wird überprüft, ob Internet vorhanden ist. Ist kein Internet verfügbar, erhält der App-Anwender einen Hinweis, dass Internet zum Abrufen des Tipps der Woche erforderlich ist. Sofern Internet vorhanden ist, wird ein neuer Tipp der Woche geladen, gespeichert und in der App angezeigt. Ruft der App-Anwender zu einem späteren Zeitpunkt die Ansicht erneut auf, wird durch eine Überprüfung ermittelt, ob ein Tipp der Woche vorhanden ist und sofern dies zutrifft, ob der vorhandene Tipp der Woche älter, als sieben Tage ist. Ist dies nicht der Fall, wird weiterhin der bestehende Tipp der Woche angezeigt. Ist der Tipp der Woche älter, als sieben Tage und Internet ist verfügbar, wird ein neuer Tipp der Woche über eine entsprechende Schnittstelle (vgl. Kapitel 5.5.2) geladen, gespeichert und angezeigt.

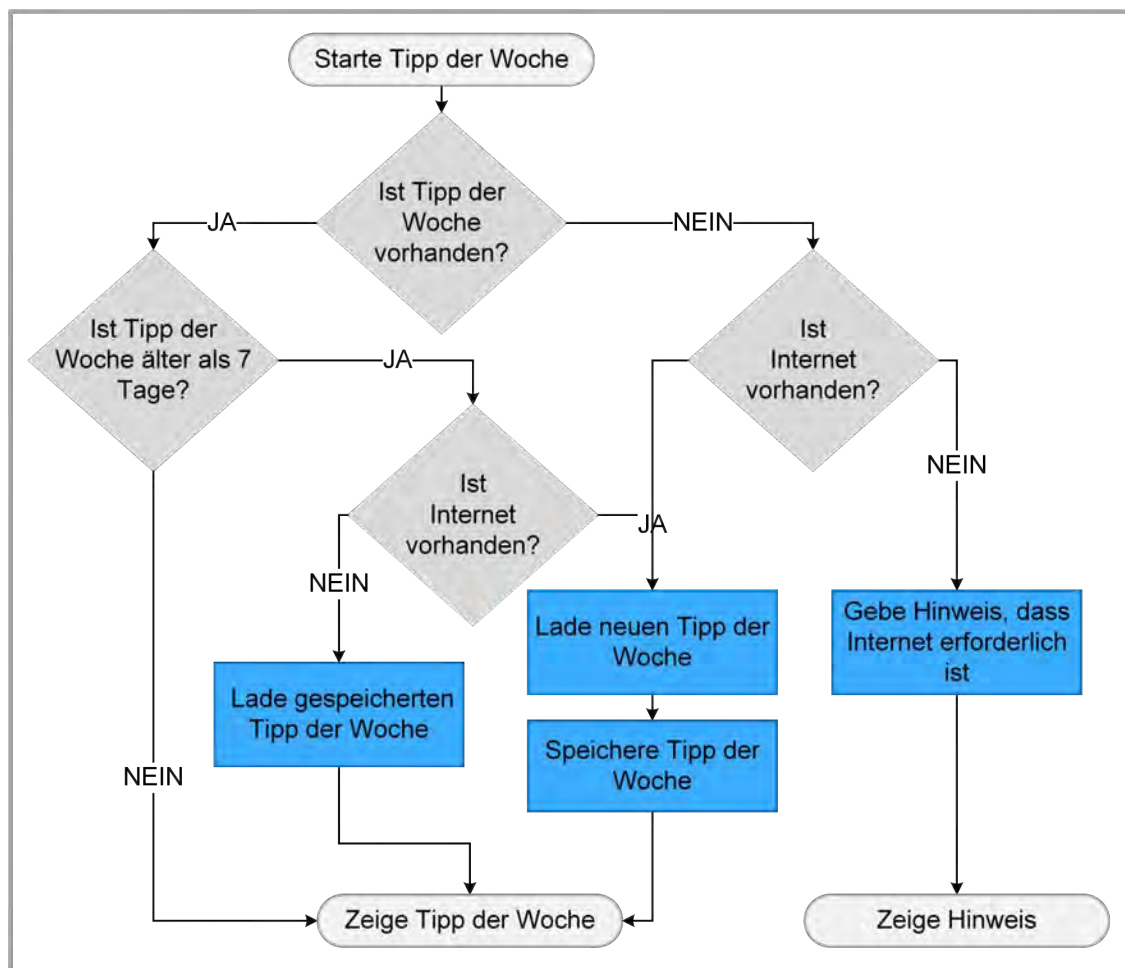


Abbildung 5.11.: Ablaufdiagramm - Tipp der Woche

Das Laden eines neuen Feedbacks als Tipp der Woche erfolgt nach dem selben Prinzip, wie es in Kapitel 5.4.1 beschrieben ist, unterscheidet sich aber in der Schnittstelle, die verwendet wird. Die Schnittstelle zum Tipp der Woche übergibt als Rückgabe-Format ein einzelnes Feedback

als JSON-Objekt. Anschließend wird wie bei den Einflussfaktoren die Methode *finishedLoading* (vgl. Listing 5.8) aufgerufen, die aus dem erhaltenen JSON-Objekt ein Feedback-Objekt erstellt (Zeile 3) und dieses in der Ansicht *TippoftheWeekFragment* anzeigt (Zeile 8). Außerdem wird der Tipp der Woche und das aktuelle Datum in den *SharedPreferences* gespeichert, damit dieses Feedback bis zur nächsten Woche nicht neu geladen werden muss.

Listing 5.8: Tipp der Woche anzeigen

```

1 public void finishedLoading(JSONObject tippoftheweek) {
2     // Create FeedbackObject from JSONObject
3     feedback.createFeedbackFromJSON(tippoftheweek);
4     ...
5     // Get Views
6     TextView title = (TextView) view.findViewById(R.id.tiptitle);
7     ...
8     title.setText(feedback.name);
9     ...
10 }

```

5.4.3. Feedback bewerten

Feedbacks können in der Detailansicht der Einflussfaktoren und in der Ansicht Tipp der Woche bewertet werden (vgl. Abbildung 5.10). Hat ein App-Anwender eine Bewertung ausgewählt, muss diese über die entsprechende Schnittstelle (vgl. Kapitel 5.5) an die TYT-Web-Anwendung gesendet werden. Dazu wird aus der Feedback-ID und der Bewertung ein JSON-Objekt erstellt. Listing 5.9 zeigt, wie die Feedback-Bewertung als JSON-Objekt in einen InputStream (Zeile 5) umgewandelt wird und zusammen mit den Anmeldedaten des App-Anwenders an die REST-Schnittstelle (vgl. Kapitel 5.5.3) als Parameter übergeben wird.

Listing 5.9: Feedback-Bewertung senden

```

1 private void sendEvaluation(JSONObject evaluation) {
2     ...
3     // Create inputStream from JSONObject
4     String evaluationString = evaluation.toString();
5     InputStream is =
6         new ByteArrayInputStream(evaluationString.getBytes());
7     RequestParams params = new RequestParams();
8     params.put("access_token", accessToken);
9     params.put("data", evaluationString);
10    EvaluationUpload asynAnswers = new EvaluationUpload(this);
11    asynAnswers.execute(new RequestParams[] { params });
12 }

```

5.4.4. Feedback geben

Der App-Anwender kann selbst Feedback geben, indem sein Feedback per E-Mail weitergeleitet wird. Dazu kann er beliebigen Text verfassen, beispielsweise indem er beschreibt, was ihm im Umgang mit dem Tinnitus hilft. Diese Rückmeldungen der App-Anwender lassen sich zukünftig dazu verwenden, um die gegebenen Feedbacks zu verbessern oder zu erweitern. Ferner kann er auch Feedback zur App geben.

5.5. Schnittstellen

Zur Kommunikation der Android-App und zukünftig auch weiteren Applikationen, mit den Komponenten Hauptgruppen-Algorithmus und Regelauswertung, sind Schnittstellen erforderlich, die den Feedback-Aufruf ermöglichen. Dazu wird die vorhandene REST-API von TrackYourTinnitus um drei Schnittstellen erweitert. Bei den Einflussfaktoren und dem Tipp der Woche kommen REST-ähnliche Web Services zum Einsatz, da die Schnittstellen der Web-Anwendung nicht direkt die Ressourcen zurückgeben, sondern erst Funktionen des Hauptgruppen-Algorithmus und der Regelauswertung ausführen und anhand der Ergebnisse Feedbacks liefern. Die dritte Schnittstelle dient dem Anlegen und Abfragen von Feedback-Bewertungen in der Datenbank. Alle drei Schnittstellen der REST-API benötigen einen *Access-Token*² als Übergabeparameter, um den App-Anwender zu identifizieren.

5.5.1. Einflussfaktoren

Die Schnittstelle, die der App die Feedbacks für die Einflussfaktoren bereitstellt, ist unter der Adresse *"api/maingroups"* über die *GET-Methode* erreichbar. Hier wird eine Liste von Feedbacks, die in den Einflussfaktoren dargestellt werden sollen, ermittelt und als eine Sammlung von JSON-Objekten zurückgegeben. Im ersten Schritt wird der Anwender über den Access-Token ermittelt. Anschließend wird der Hauptgruppen-Algorithmus (vgl. Kapitel 5.6.1) für den App-Anwender gestartet, der die richtigen Hauptgruppen in die Datenbank speichert. Anhand der ermittelten Hauptgruppen liefert die Methode *MaingroupHelper::get_maingroupfeedbacks*, des Hauptgruppen-Algorithmus, ein Array von Feedbacks zurück, die für den App-Anwender zutreffen. Dieses Array wird in der Schnittstelle in ein JSON-Objekt umgewandelt und an die App übertragen. Listing 5.10 zeigt das Format, in dem das JSON-Objekt übertragen wird. Dieses enthält neben den Feedbacks zu den vier Einflussfaktoren auch den Farbwert der DOR-Werte, damit die App die Datenzuverlässigkeit farblich entsprechend anzeigen kann.

Listing 5.10: Ausgabeformat der Schnittstelle "Einflussfaktoren"

```
1  JSON-Object {
2      "stress" : [ DOR-Farbwert-Stress, { Feedback-Objekt }, ... ],
3      "emotional" : [ DOR-Farbwert-Emotional, {}, ... ],
4      "arousal" : [ DOR-Farbwert-Aufregung, {}, ... ],
5      "concentration" : [ DOR-Farbwert-Konzentration, {}, ... ]
6  }
```

²Bei einem Access-Token handelt es sich um verschlüsselte Anmeldeinformationen des App-Anwenders.

Optional kann diese Schnittstelle auch ein Start- und ein End-Datum entgegennehmen. In diesem Fall wird der Hauptgruppen-Algorithmus mit diesen als Parameter gestartet.

5.5.2. Tipp der Woche

Die Schnittstelle Tipp der Woche ist unter der Adresse "*api/tippoftheweek*" über die *GET-Methode* erreichbar. Sie liefert der App ein Feedback zurück, das diese dem App-Anwender als Tipp der Woche anzeigen kann.

Beim Tipp der Woche handelt es sich nicht um eine Ressource, die aus der Datenbank abgerufen wird, sondern es wird ein spezielles Feedback erwartet. Die Datenbasis für den Tipp der Woche bilden alle Feedbacks aller möglichen Feedback-Arten (vgl. Tabelle 4.2), die für den App-Anwender zutreffen – dies beinhaltet Feedbacks aus den Einflussfaktoren. Dies bedeutet es werden alle Hauptgruppen-Feedbacks, denen der App-Anwender angehört sowie alle individuellen Feedbacks, deren Regel der App-Anwender erfüllt (vgl. Kapitel 5.6.2), in ein Array geschrieben (vgl. Kapitel 5.6.2). Alle Feedbacks, die sich in diesem Array befinden können als Tipp der Woche gegeben werden.

Beim Aufruf des Tipp der Woche, wird ausgehend von der App, innerhalb der Schnittstelle die Methode *get_tippoftheweek()* aufgerufen. Diese Methode fordert zum einen von der Klasse *TippoftheweekHelper* alle Feedbacks für einen App-Anwender durch Aufruf der Methode *getFeedbackForUser(\$userid)* an und ruft sofern Feedbacks für einen App-Anwender vorhanden sind die Methode *getNewFeedback(\$userid, \$feedbacks)* in der der Klasse *TippoftheweekHelper* zur Ermittlung eines einzelnen Feedbacks auf, das der App geliefert wird.

Jedoch soll das Feedback, das im Tipp der Woche angezeigt wird jede Woche ein anderes sein. Dazu werden die IDs der angezeigten Feedbacks in der Tabelle „usermemories“ für den App-Anwender gespeichert, um diese Feedbacks in künftigen Abfragen auszuschließen. Sobald das Array der möglichen Feedbacks lediglich noch Feedbacks enthält, die bereits in der Tabelle vermerkt sind, wird die Tabelle „usermemories“ für den App-Anwender zurückgesetzt und der Ablauf beginnt erneut.

Das als Tipp der Woche ermittelte Feedback wird als JSON-Objekt, das alle Werte des Feedbacks aus der Datenbanktabelle enthält, an die App zurückgeben. Das Format des JSON-Objekt ist in Listing 5.11 veranschaulicht.

Listing 5.11: Ausgabeformat der Schnittstelle "Tipp der Woche"

```

1  JSON-Object {
2      "title" : "Feedback Titel",
3      "description" : "Feedback Beschreibung",
4      "short_description" : "Kurzbeschreibung",
5      ...
6  }
```

5.5.3. Feedback-Bewertung

Zum Bewerten von Feedbacks gibt es eine Schnittstelle, um Feedback-Bewertungen in der Datenbank anzulegen. Diese ist unter der Adresse "*api/feedbackevaluation*" über die *POST-*

Methode erreichbar und nimmt eine Feedback-Bewertung als JSON-Objekt entgegen. Das Format des JSON-Objekts ist in Listing 5.12 dargestellt.

Listing 5.12: Eingabeformat der Schnittstelle "Feedback-Bewertung"

```
1  JSON-Object {
2    "feedback-id" : "1",
3    "helpful" : "true"
4  }
```

Die Feedback-Bewertung wird in der Datenbanktabelle "feedbackevaluations" (vgl. Kapitel 5.2) mit der Feedback-ID, der App-Anwender-ID und der Bewertung angelegt. Listing 5.13 zeigt, dass die Schnittstelle das JSON-Objekt zunächst dekodiert (Zeile 2), d.h. es in ein PHP-Objekt umwandelt und dessen Eigenschaften "feedback_id" und "helpful" dem Feedback-Bewertungs-Objekt übergibt, sofern diese vorhanden sind (Zeile 7-11). Anschließend wird die Feedback-Bewertung durch die save-Methode von Eloquent ORM in der Datenbank gespeichert.

Listing 5.13: Feedbackbewertung speichern

```
1  // decode JSON-object
2  $data = json_decode($data_json);
3  // Create new Feedbackevaluation
4  $feedbackevaluation = new FeedbackEvaluation();
5  $feedbackevaluation -> user_id = $user['id'];
6  // check whether the data is complete
7  if(property_exists($data, 'feedback_id')) {
8      $feedbackevaluation -> feedback_id = $data->feedback_id;
9  }
10 if(property_exists($data, 'helpful')) {
11     $feedbackevaluation -> helpful = $data->helpful;
12 }
13 // Save Feedbackevaluation to Database
14 $feedbackevaluation->save();
```

5.6. Feedback-Algorithmus

Im Folgenden wird die Implementierung des Feedback-Algorithmus erläutert. Dabei wird beschrieben, wie der Algorithmus zur Ermittlung der Hauptgruppen abläuft und wie die zutreffenden Feedbacks ermittelt werden, die über die Schnittstellen aufgerufen werden (vgl. Kapitel 5.5). Außerdem wird die Implementierung der Regelauswertung erläutert. Zu diesem Zweck werden die beiden PHP-Klassen *MaingroupHelper*, für den Hauptgruppen-Algorithmus und *TippoftheweekHelper* entwickelt. Die Klasse *TippoftheweekHelper* enthält Funktionen zur Ermittlung des Tipp der Woche sowie zur Regelauswertung.

5.6.1. Hauptgruppen-Algorithmus

Der Hauptgruppen-Algorithmus wird von der Schnittstelle Einflussfaktoren (vgl. Kapitel 5.5.1) initiiert, indem diese die Funktion *updateMaingroups(\$userid, \$dateStart, \$dateEnd)* für einen App-Anwender aufruft und dessen Benutzer-ID übergibt. Neben der Benutzer-ID wird die Funktion entweder zusätzlich mit einem Start- und Enddatum (vgl. Kapitel 5.4.1) aufgerufen oder andernfalls ist für die beiden Parameter kein Wert gesetzt. Der Ablauf des implementierten Hauptgruppen-Algorithmus ist in Abbildung 5.12 dargestellt.

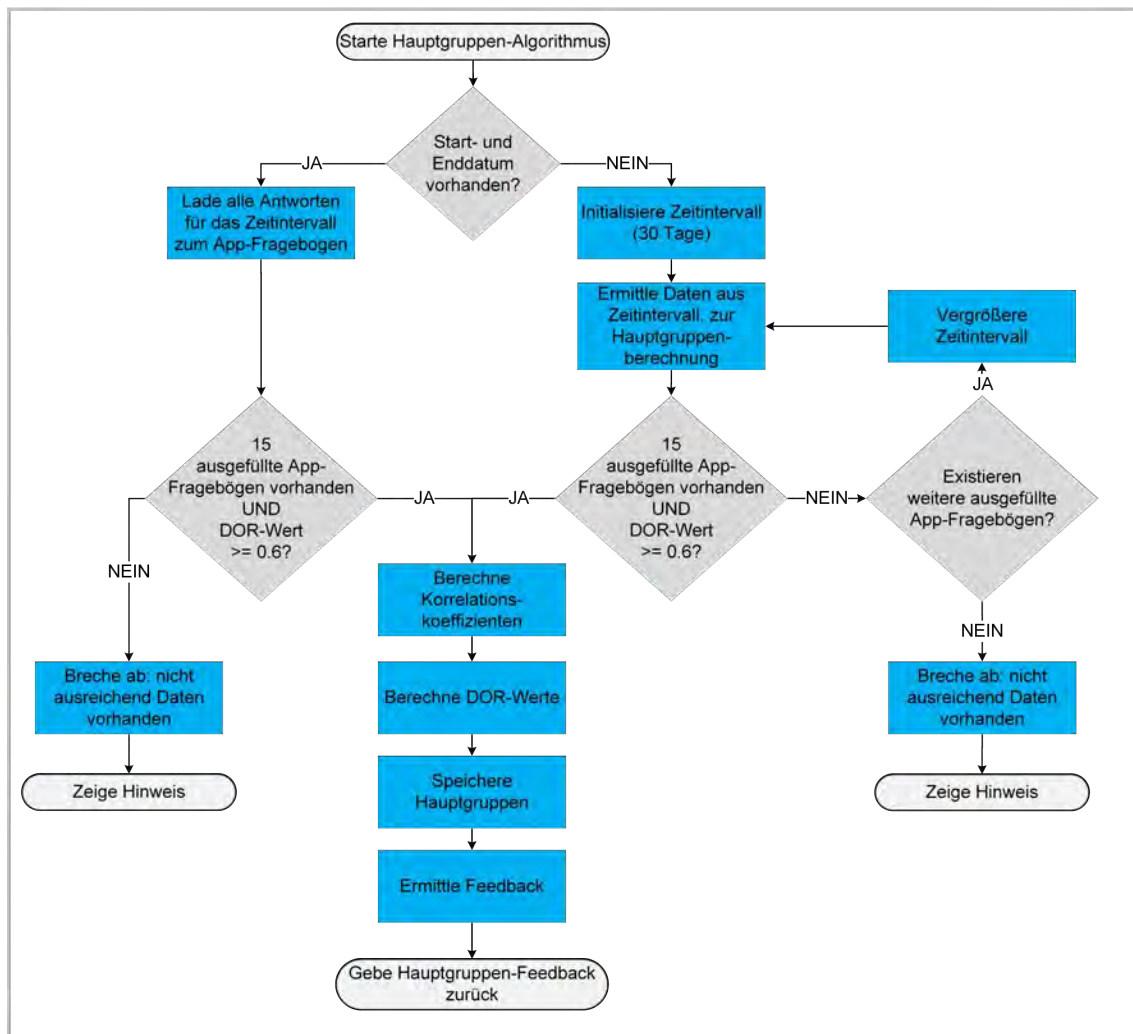


Abbildung 5.12.: Ablaufdiagramm - Hauptgruppen-Algorithmus

Schritt 1: Ermittlung des Datenzeitraums

Beim ersten Aufruf des Hauptgruppen-Algorithmus ist standardmäßig keine Datumsangabe vorhanden. Es muss daher ermittelt werden, für welches Zeitintervall Daten für die Berechnung der Hauptgruppen einbezogen werden. Listing 5.14 zeigt die Implementierung der Ermittlung des Datenzeitraums. Dazu wird zunächst ausgehend vom aktuellen Datum ermittelt, an

welchem Datum der erste App-Fragebogen ausgefüllt worden ist und wie viele Tage zwischen dem ersten ausgefüllten App-Fragebogen und dem aktuellsten liegen (Zeile 2-3). Diese ermittelte Anzahl an Tagen dient als Grenze, um zu verhindern, dass der Algorithmus beim Laden der Antworten zum App-Fragebogen zeitlich beliebig weit zurück geht. Die Grenze erlaubt es ihm maximal bis zu diesem Datum des ersten ausgefüllten App-Fragebogen die entsprechenden Antworten des App-Anwenders zu laden, da davor keine existieren.

Listing 5.14: Hauptgruppen-Algorithmus - Ermittlung des Datenzeitraums

```

1  // Get oldest answer: loop will end if this date is reached
2  $oldestanwer = Standardanswer::where('user_id', '=', $userid)->
    order_by('created_at', 'asc')->first()->original['created_at'];
3  $maxDateinterval = Maingrouphelper::dateDifference($oldestanwer);
4  $dateinterval = 30;
5  $loop = true;
6  // Loop till a DOR-Value is high enough for feedback
7  do {
8      // check if the loop should again, after this
9      if($dateinterval>$maxDateinterval){
10         $loop = false;
11     } // Calculate Date interval to collect data from
12     $date = new DateTime;
13     $date->modify('-'. $dateinterval .'days');
14     $formatted_date = $date->format('Y-m-d H:i:s');
15
16     $patientanswers = Standardanswer::where('user_id', '=', $userid
        )->where('created_at', '>=', $formatted_date) -> get();
17     if (empty($patientanswers) || $patientanswers == "" || is_null(
        $patientanswers)){
18         // patient hasn't entered data for a month, back 30 days
19         $dateinterval = $dateinterval +30;
20     }
21     else{
22         if(Maingrouphelper::dorIsHighEnough($patientanswers, $userid)
            && sizeof($patientanswers) > 14){
23             Maingrouphelper::writeMaingroupsFromPatientanwers(
                $patientanswers, $userid);
24             return true;
25         }
26         else{
27             // dor values are to low: go back 10 days
28             $dateinterval = $dateinterval + 10;
29         }
30     }
31     // if dateintervall is higher than the last answer given by user,
32     // there is no DOR-Value high enough. return false
33 } while ($loop);

```

Nachdem die Grenze bekannt ist wird überprüft, ob ausgehend vom Datum des aktuellsten Datensatzes innerhalb der letzten 30 Tage, wie im Konzept definiert (vgl. Kapitel 4.4.1), mindestens 15 ausgefüllte App-Fragebögen vorhanden sind und mindestens für einen Einflussfaktor die Daten zuverlässig sind (Zeile 22). Hierbei lassen sich drei Fälle unterscheiden:

- Fall 1: Der erste ausgefüllte App-Fragebogen liegt im Zeitintervall der definierten letzten 30 Tage, aber innerhalb des Zeitintervalls existieren keine 15 ausgefüllte App-Fragebögen. Dieser Fall kommt dann vor, wenn es sich um einen neu am System angemeldeten App-Anwender handelt, der zwar innerhalb des letzten Monats App-Fragebögen ausgefüllt hat, aber nicht genügend. Dies stellt ein Abbruchkriterium für den Algorithmus dar (Zeile 9). Über die Schnittstelle wird der App mitgeteilt, dass zu wenig Daten vorhanden sind.
- Fall 2: Der erste ausgefüllte App-Fragebogen liegt entweder im Zeitintervall der definierten letzten 30 Tage oder vor diesem Zeitintervall. Innerhalb des Zeitintervalls existieren 15 ausgefüllte App-Fragebögen. Dieser Fall kann auftreten, wenn ein neu angemeldeter App-Anwender beispielsweise seit 15 Tagen täglich App-Fragebögen ausfüllt oder ein App-Anwender bereits länger angemeldet ist und innerhalb des letzten Monats ausreichend App-Fragebögen ausgefüllt hat. Daraufhin werden alle Fragebögen, die für das Zeitintervall existieren geladen und es erfolgt die Berechnung der Datenzuverlässigkeit (vgl. *Schritt 2*).
- Fall 3: Der erste ausgefüllte App-Fragebogen liegt vor dem Zeitintervall der letzten 30 Tage und innerhalb des Zeitintervalls existieren keine 15 ausgefüllte App-Fragebögen. Dieser Fall kann beispielsweise auftreten, wenn ein App-Anwender vergleichsweise zu früher innerhalb des letzten Monats keinen App-Fragebogen ausgefüllt hat. Sind innerhalb der letzten 30 Tage keine Fragebögen vorhanden, wird zeitlich solange um 30 Tage zurückgesprungen, bis ausgefüllte Fragebögen vorhanden sind. Anschließend wird in 10-Tagesschritten solange zurückgesprungen, bis mindestens 15 ausgefüllte Fragebögen erreicht und die Daten zuverlässig sind. Ist die als Abbruchkriterium definierte Datums-Grenze erreicht, wird der Hauptgruppen-Algorithmus beendet und dies der Schnittstelle mitgeteilt.

Beim Laden aller ausgefüllten App-Fragebögen eines App-Anwenders werden die Werte aller Antworten zur Tinnitus-Lautheit, Stress, Emotion, Aufregung und Konzentration in einem mehrdimensionalen Array gespeichert (vgl. Abbildung 5.13). Beim Ermitteln der beantworteten App-Fragebögen muss vor der Berechnung der Datenzuverlässigkeit überprüft werden, ob App-Fragebögen unvollständig ausgefüllt sind. Die unvollständigen Fragebögen müssen entfernt werden, sofern sie *NULL*-Werte enthalten. Wie im Beispiel in Abbildung 5.13 dargestellt, beinhaltet das Array der Tinnitus-Lautheit ("loudnessArray") an Indexstelle 5 und 6 den Wert *NULL*. Dies bedeutet, dass die entsprechenden Fragen des App-Fragebogens nicht ausgefüllt sind. Da diese Datensätze für die weiteren Berechnungen unbrauchbar sind, werden in allen Arrays an der entsprechenden Indexstelle die Datenwerte gelöscht. In diesem Fall an Indexstelle 5 und 6.

loudnessArray { [0]=> float(0.17000000178814) [1]=> float(0.17000000178814) [2]=> float(0.0099999997764826) [3]=> float(0.0099999997764826) [4]=> float(0) [5]=> NULL [6]=> NULL [7]=> float(0.029999999329448) [8]=> float(0.0099999997764826) [9]=> float(0.03999999910593) [10]=> float(0.029999999329448) ⋮	emotionalArray { [0]=> float(0.75) [1]=> float(0.75) [2]=> float(0.75) [3]=> float(0.75) [4]=> float(0.75) [5]=> float(0.75) [6]=> float(0.75) [7]=> float(0.75) [8]=> float(0.75) [9]=> float(0.75) [10]=> float(0.75) ⋮	arousalArray { [0]=> float(0.25) [1]=> float(0.25) [2]=> float(0) [3]=> float(0) [4]=> float(0) [5]=> float(0) [6]=> float(0) ⋮	stressArray { [0]=> float(0.5) [1]=> float(0.5) [2]=> float(0.30000001192093) [3]=> float(0.30000001192093) [4]=> float(0.37999999523163) [5]=> float(0.0099999997764826) [6]=> float(0.0099999997764826) ⋮	concentrationArray { [0]=> float(0.60000002384186) [1]=> float(0.60000002384186) [2]=> float(0.97000002861023) [3]=> float(0.97000002861023) [4]=> float(0.99000000953674) [5]=> float(0.95999997854233) [6]=> float(0.95999997854233) ⋮
---	---	---	---	--

Abbildung 5.13.: Beispiel - Entfernung von NULL-Werten

Schritt 2: Berechnung der Datenzuverlässigkeit

Wie in Kapitel 4.4.1 erklärt, wird der DOR-Wert anhand der Daten eines App-Anwenders aus den beantworteten App-Fragebögen berechnet. Ein Problem, das hierbei gelöst werden muss, ist die Aufteilung und Sortierung der Daten zur Tinnitus-Lautheit.

Die Werte der Tinnitus-Lautheit ("loudnessArray") werden aufsteigend der Größe nach geordnet, da dies für die Berechnung des DOR-Wertes erforderlich ist. Allerdings ist eine alleinige Sortierung der Werte der Tinnitus-Lautheit, wie in Abbildung 5.14 nicht ausreichend. Zusätzlich ist es zwingend erforderlich die ursprüngliche Indexposition des jeweiligen Wertes der anderen Einflussfaktoren zu kennen. Realisiert ist dies als Key-Value-Paar. An erster Indexposition des sortieren Arrays ("loudnessArray") befindet sich das Key-Value-Paar (4 - 0). Wobei der *Key* 4 die ursprüngliche Indexposition darstellt (siehe grüne Markierung in Abbildung 5.13). Da für die Berechnung von Korrelationskoeffizienten und DOR-Werten für die Einflussfaktoren die Werte der Tinnitus-Lautheit ("loudnessArray") mit den zugehörigen Werten aus den anderen Einflussfaktoren-Arrays in Beziehung gesetzt werden müssen, bedeutet dies, dass die restlichen Arrays nicht, wie die Werte der Tinnitus-Lautheit aufsteigend sortiert

werden können. Die Sortierung der anderen Arrays orientiert sich am "loudnessArray", um den richtigen Wert zu finden. Das Vorgehen wird am Beispiel des Stress-Arrays ("stressArray") erläutert. Für jeden Wert aus dem "loudnessArray", wird ein Key verwendet, um die Position im "stressArray" zu ermitteln. Demnach wird der Wert an Indexposition 4 im "stressArray" (siehe blaue Markierung in Abbildung 5.13) an die erste Position gesetzt, der Wert an Indexposition 9 an die zweite Position usw. bis alle Werte des "stressArray" entsprechend der Sortierung des "loudnessArray" sortiert sind. Dieses Vorgehen wird auf die anderen Einflussfaktoren übertragen.

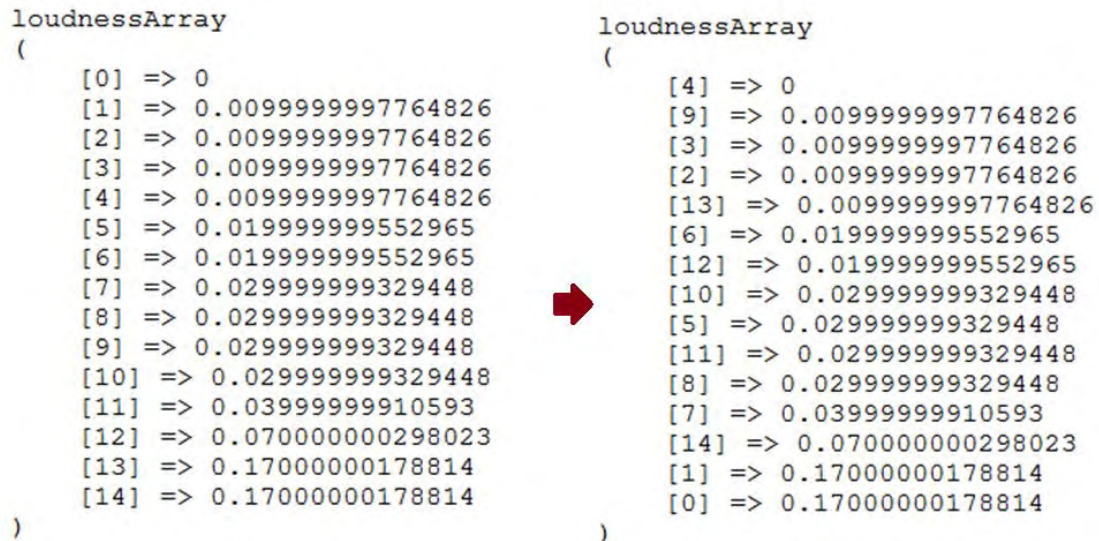


Abbildung 5.14.: Beispiel - Sortierung der Datenwerte der Tinnitus-Lautheit

Abbildung 5.15 veranschaulicht die Datenwerte des sortierten "loudnessArray" und deren Aufteilung in zwei Gruppen (Split-Half-Verfahren). Hierbei werden die Datenwerte gleichmäßig in die zwei Arrays "loudnessGroup1Array" und "loudnessGroup2Array" geschrieben. Die anderen sortierten Einflussfaktoren-Arrays werden auf die selbe Weise in zwei Gruppen aufgeteilt. Beispielsweise werden für das "stressArray" die zwei Arrays "stressGroup1Array" und "stressGroup2Array" angelegt.

Die Berechnung der DOR-Werte erfolgt, wie in Konzept 4.4.1 beschrieben, indem für alle Einflussfaktoren jeweils zwei Korrelationskoeffizienten aus den aufgeteilten Arrays berechnet werden. Am Beispiel Stress werden die Korrelationskoeffizienten zwischen "loudnessGroup1Array" und "stressGroup1Array" sowie zwischen "loudnessGroup2Array" und "stressGroup2Array" berechnet. Anschließend wird zur Berechnung des DOR-Wertes der kleinere Korrelationskoeffizient ermittelt und durch den größeren geteilt. Nach demselben Prinzip erfolgt die Vorgehensweise für die anderen Einflussfaktoren.

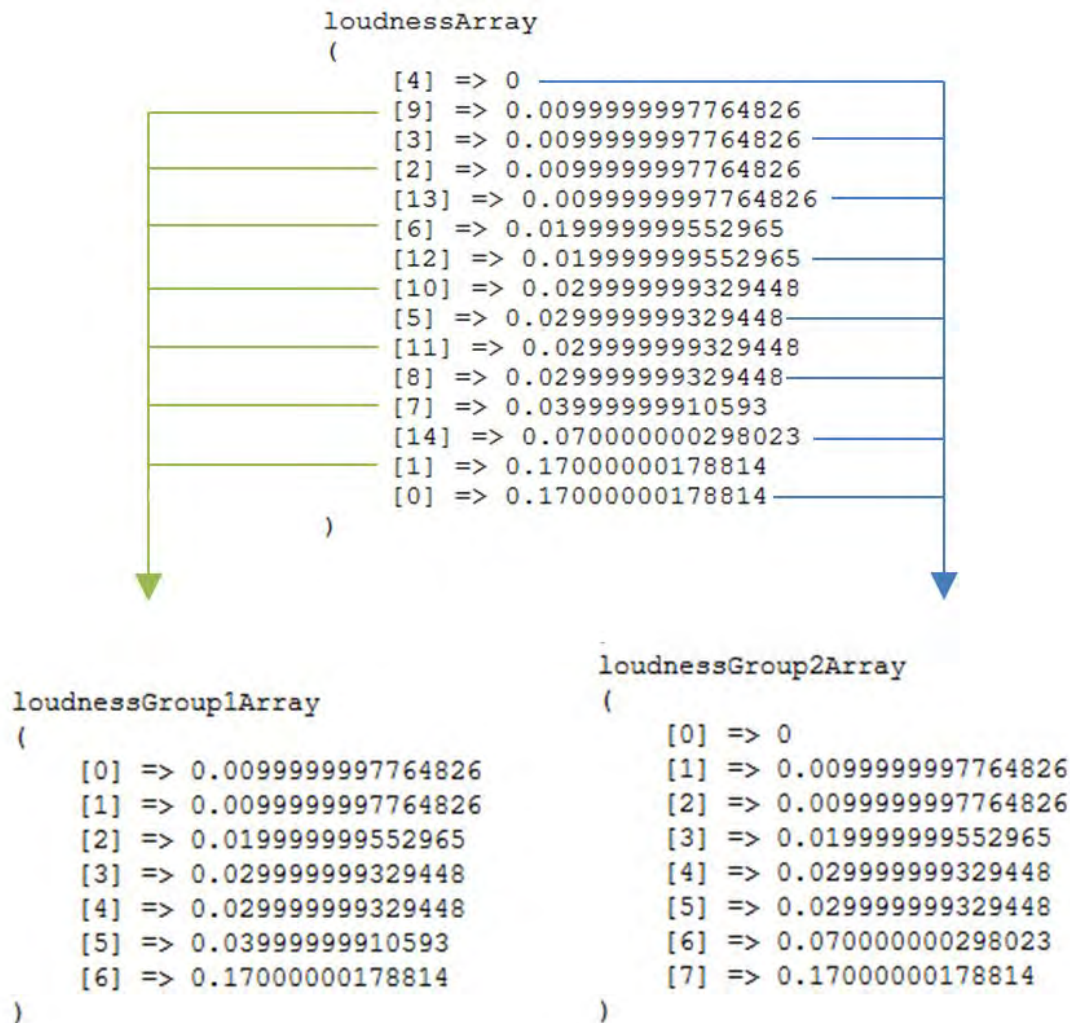


Abbildung 5.15.: Beispiel - DOR-Berechnung durch Aufteilung der Datenwerte der Tinnitus-Lautheit

Schritt 3: Prüfung der Datenzuverlässigkeit

Dieser Schritt prüft die vier ermittelten DOR-Werte der Einflussfaktoren aus *Schritt 2*. Die Daten gelten dabei als verlässlich, wenn einer der DOR-Werte größer als 0.6 ist (vgl. Kapitel 4.4.1).

Schritt 4: Ermittlung der Hauptgruppen

Nachdem der Datenzeitraum ermittelt worden ist und die Daten als zuverlässig gelten, kann in diesem Schritt die Berechnung der Hauptgruppen beginnen. Dazu wird die Funktion *writeMaingroupsFromPatientanswers(\$patientanswers, \$userid)* aufgerufen. Diese erwartet neben der ID des App-Anwenders ein Array (*\$patientanswers*) mit allen gültigen App-Fragebögen (vgl. *Schritt 2*), die zur Berechnung verwendet werden sollen. Die Implementierung dieser Funktion ist in Listing 5.16 dargestellt. Der erste Schritt der Funktion ist die Aufbereitung der beantworteten Fragebögen. Hierzu wird, wie bei der DOR-Wert-Berechnung in *Schritt 2* ein multidimensionales Array initialisiert (Zeile 3-9), das die Datenwerte der

Einflussfaktoren aller beantworteten App-Fragebögen jeweils in einem Array speichert. Das übergebene Array (*\$patientanswers*) wird in einer Schleife durchlaufen (Zeile 12-23).

Nachdem alle Einflussfaktoren in das multidimensionale Array eingeordnet sind, werden die Korrelationskoeffizienten (vgl. Kapitel 4.2) der vier Einflussfaktoren: Stress, Konzentration, Emotion und Aufregung jeweils in Zusammenhang mit der Tinnitus-Lautheit berechnet (Zeile 25-32). Aus den berechneten Korrelationskoeffizienten werden die Hauptgruppen ermittelt und in die Datenbank geschrieben (Zeile 36). Anschließend werden die DOR-Werte aller Einflussfaktoren und die berechneten Korrelationskoeffizienten des App-Anwenders in der Tabelle *maingroupuservvalues* (vgl. Kapitel 5.2) aktualisiert. Hierbei wird zunächst geprüft, ob der App-Anwender bereits einen Eintrag in dieser Tabelle hat oder ob ein neuer Eintrag für ihn angelegt werden muss (Zeile 43-54). Das Speichern der Korrelationskoeffizienten und DOR-Werte in der Datenbank erfolgt, wie in Kapitel 5.3 beschrieben über *Eloquent ORM* von Laravel.

Listing 5.15: Hauptgruppen-Algorithmus - Werte berechnen

```

1 public static function writeMaingroupsFromPatientanswers(
    $patientanswers, $userid){
2     //Multidimensional array with values of each question
3     $calc = array(,
4         "loudness" => array(),
5         "emotional" => array(),
6         "arousal" => array(),
7         "stress" => array(),
8         "concentration" => array(),
9     );
10    ...
11    // Loop through all patientanswers: put them in calc-Array
12    foreach($patientanswers AS $standardanswerobject){
13        ...
14        foreach($standardanswerobject->attributes AS $a){
15            // TINNITUSLOUDNESS (question2)
16            if ($i == 3)
17                array_push($calc["loudness"], $a);
18            // EMOTIONAL (question4)
19            if ($i == 5)
20                array_push($calc["emotional"], $a);
21            ...
22        }
23    }
24    //Calculate calcCorrelationValues
25    $corrLoudnessEmotional = Maingrouphelper::calculateCorrelation
26        ($calc["loudness"], $calc["emotional"]);
27    $corrLoudnessArousal = Maingrouphelper::calculateCorrelation
28        ($calc["loudness"], $calc["arousal"]);
29    $corrLoudnessStress = Maingrouphelper::calculateCorrelation
30        ($calc["loudness"], $calc["stress"]);

```

```

31  $corrLoudnessConcentration =
32      Maingrouphelper::calculateCorrelation
33          ($calc["loudness"], $calc["concentration"]);
34
35  // Save Maingroups in Database
36  Maingrouphelper::updateUserMaingroups($userid,
      $corrLoudnessStrain, $corrLoudnessStress,
      $corrLoudnessEmotional, $corrLoudnessArousal,
      $corrLoudnessConcentration);
37
38  // Calculate DOR Values
39  $dorLoudnessStress = Maingrouphelper::calculateDOR
40      ($calc["loudness"], $calc["stress"]);
41  ...
42  // save everything in Database
43  $currentDataSet = new MaingroupUservalues();
44  try{
45      $currentDataSet = MaingroupUservalues::where
46          ('user_id', '=', $userid) -> first();
47      if (is_null($currentDataSet)){
48          $currentDataSet = new MaingroupUservalues();
49      }
50  }
51  catch(Exception $e){}
52  $currentDataSet -> user_id = $userid;
53  ...
54  $currentDataSet -> save();
55  }

```

Schritt 5: Feedback-Rückgabe

Über die in *Schritt 4* ermittelten Hauptgruppen kann dem App-Anwender Feedback gegeben werden, das für ihn zutrifft. Zu diesem Zweck implementiert der Hauptgruppen-Algorithmus die in Listing 5.16 dargestellte Funktion. Diese Funktion lädt zunächst aus der Datenbank die in *Schritt 4* ermittelten Hauptgruppen und DOR-Werte, des Tinnitus-Betroffenen (Zeile 3, 14). Um das definierte Schnittstellen-Format zur Übergabe von Einflussfaktoren einzuhalten (vgl. Kapitel 5.5.1), wird ein multidimensionales Array initialisiert, das für jeden Einflussfaktor ein Array enthält. An der erster Position des Arrays befindet sich der korrespondierende DOR-Wert (Zeile 7-11).

Aus der Datenbank werden alle Feedbacks abgefragt, die der Hauptgruppe des App-Anwenders entsprechen und anschließend werden sie in das entsprechende Array geschrieben (Zeile 20-28). Dabei werden ausschließlich Hauptgruppen-Feedbacks berücksichtigt. Sobald alle zutreffenden Feedbacks in die entsprechenden Arrays geschrieben sind, wird das multidimensionale Array an die Schnittstelle zurückgegeben.

Listing 5.16: Hauptgruppen-Algorithmus - Feedback vergeben

```

1 public static function get_maingroupfeedbacks($userid) {
2     // Get the dor-values for the user
3     $currentDataSet = MaingroupUservalues::where
4         ('user_id', '=', $userid) -> first() ->
5         original;
6
7     // initlize array with dor
8     $feedbacks = array(
9         "stress" => array(Maingrouphelper::getDORColor(
10             $currentDataSet["dor_stress"])),
11         "emotional" => array(Maingrouphelper::getDORColor(
12             $currentDataSet["dor_emotional"])),
13         "arousal" => array(Maingrouphelper::getDORColor(
14             $currentDataSet["dor_arousal"])),
15         "concentration" => array(Maingrouphelper::getDORColor(
16             $currentDataSet["dor_concentration"]));
17
18     // Get the maingroups of the user from the db
19     $userGroups= UserFeedbackGroups::where
20         ('user_id', '=', $userid) -> first() -> original;
21
22     $mgStressID = $userGroups['mg_stress'];
23     ...
24     // Get Feedback for the maingroups
25     $stressFeedbacks = Feedback::where('maingroup', '=',
26         $mgStressID ) -> get();
27     ...
28     // loop through all arrays, and add the maingroupfeedbacks
29     // (the ones without subgroup_rules) to the resultarray
30     foreach($stressFeedbacks AS $f){
31         if ($f->original['sg_rule'] == ""){
32             array_push($feedbacks["stress"], $f->original);
33         }
34     }
35     ...
36     return $feedbacks;
37 }

```

5.6.2. Regelauswertung

Zur Ermittlung von Feedbacks für den Tipp der Woche erfolgt eine Regelauswertung. Die Regelauswertung erfolgt innerhalb der Klasse *TippoftheweekHelper*, nachdem die Schnittstelle Tipp der Woche (vgl. Kapitel 5.5.2) Feedback für einen App-Anwender anfordert. Der Ablauf zur Ermittlung von Feedback für den Tipp der Woche ist in Abbildung 5.16 dargestellt.

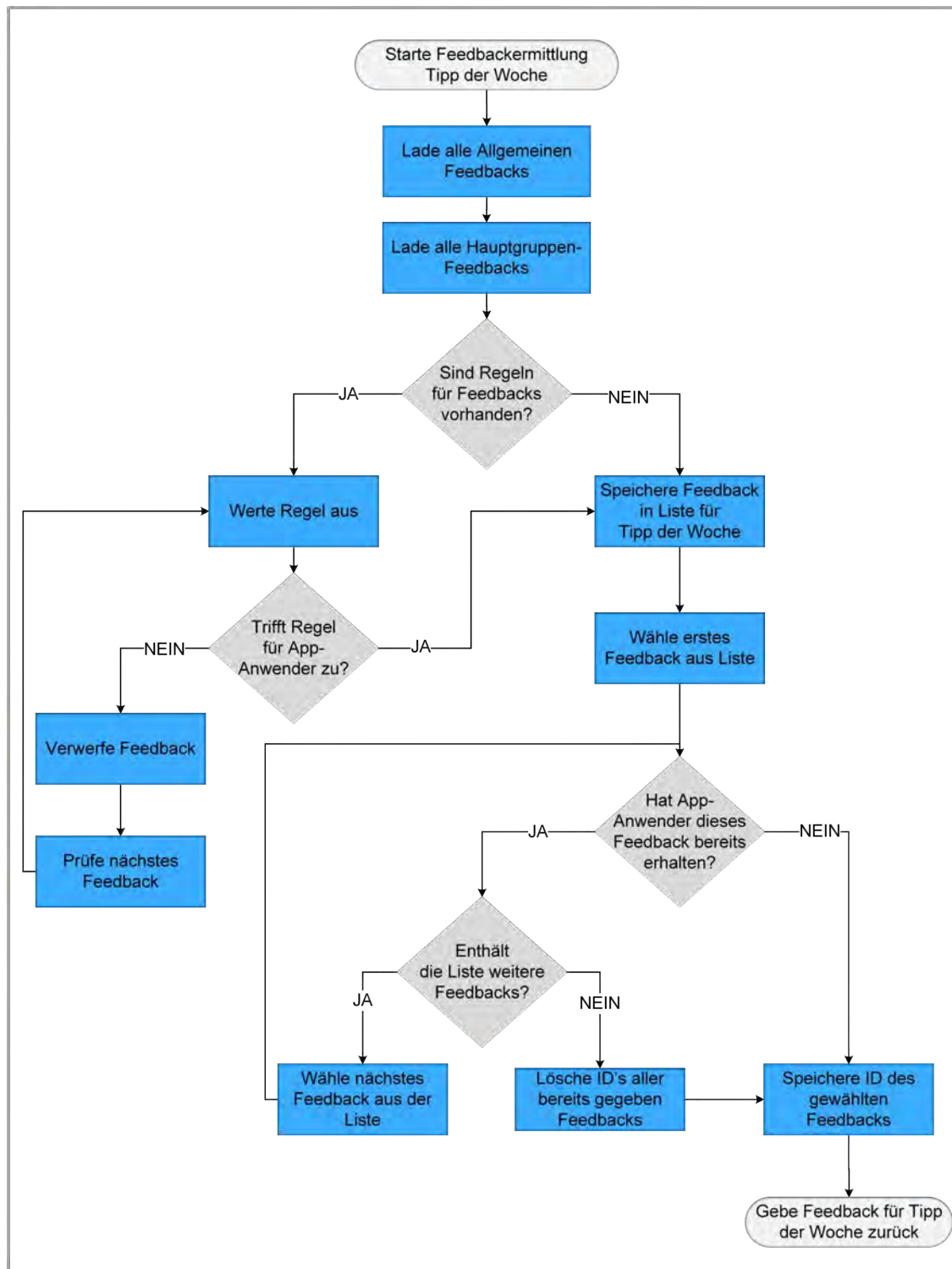


Abbildung 5.16.: Ablaufdiagramm - Feedback-Ermittlung

Innerhalb der Funktion *getFeedbackForUser(\$userid)* werden für einen App-Anwender zunächst alle allgemeinen Feedbacks aus der Datenbank geladen, die für alle App-Anwender gültig sind. Anschließend werden alle existierenden Feedbacks zu den Hauptgruppen geladen, denen der App-Anwender zugewiesen ist. Für jedes geladene Feedback wird geprüft, ob eine Regel vorhanden ist. Sofern ein Feedback keine Regel aufweist, wird es als möglicher Tipp der Woche zwischengespeichert. Verfügt ein Feedback über eine Regel, wird für alle Feedbacks ermittelt, ob die Regelbedingungen für den App-Anwender zutreffen und erst wenn dies der Fall ist, wird das entsprechende Feedback ebenfalls zwischengespeichert. Andernfalls bedeutet dies, dass die Regel für den App-Anwender nicht zutrifft und das Feedback für ihn nicht gegeben werden kann.

Die Regeln zur Regelauswertung sind in Tabelle „feedbacks“ im Feld „rule“ definiert, wie im Konzept (vgl. Kapitel 4.4) beschrieben ist. Das Überprüfen, ob eine Regel für den App-Anwender zutrifft, erfolgt durch die Funktion *checkRuleForUser(\$userid, \$rules, \$fields)*, die in Listing 5.17 dargestellt ist.

Listing 5.17: Regelauswertung

```

1 public static function checkRuleForUser($userid, $rules, $fields)
2 {
3     // split the fields string to array
4     $fields = explode(",", $fields);
5
6     // get the answers of the user to the given fields
7     foreach($fields AS $field){
8         // get the answer of the user to the question with id==field
9         $answer = Answer::where('user_id', '=', $userid )
10             ->where('question_id', '=', $field )->get();
11         if (sizeof($answer) > 0){
12             $answer = $answer[sizeof($answer)-1]->original;
13             // get the answer-field from the dataset
14             $answer = "'" . $answer['answer'] . "'";
15
16             // replace the answer id (placeholder) with the
17             // corresponding answer-value
18             $rules = str_replace($field, $answer, $rules);
19         }
20         // if the size of $answer is 0,
21         // the user hasnt answered that question, so return false
22         else{
23             return false;
24         }
25     }
26     // now $rules that mapps the answers of the user with the rules
27     // we evaluate the string and return the result
28     $result = eval("return " . $rules . ";");
29     return $result;
30 }

```

Die Funktion erhält drei Parameter. Die ID des App-Anwenders (\$userid), die zu überprüfende Regel (\$rules) und die IDs der Fragen (\$fields) aus den Fragebögen, die innerhalb der Regel überprüft werden. Zunächst werden zu allen Fragen, die innerhalb der Regel überprüft werden, alle Antworten des App-Anwenders aus der Datenbank geladen. Im Regel-String (\$rules) werden anschließend die IDs der Fragen, die in der Regel vorkommen, durch die Antwort des App-Anwenders ersetzt (Zeile 17), die er zu diesen Fragen gegeben hat. Sind zu allen definierten Fragen innerhalb einer Regel, alle Antworten des App-Anwenders in den Regel-String eingesetzt, wird die Regel durch die eval-Funktion (Zeile 27) überprüft. Das Ergebnis der eval-Funktion gibt an, ob die Regel für den App-Anwender zutrifft oder nicht.

Abbildung 5.17 zeigt, wie eine definierte Regel, die durch einen Administrator in der Datenbank angelegt ist aussehen kann und visualisiert, wie diese Regel ausgewertet wird. Dazu wird die Regelauswertung zu Option 1 aus Kapitel 5.3 beispielhaft dargestellt. Die erstellte Regel beinhaltet die Datenbank-IDs zu den Fragen der Fragebögen. Die Variable \$fields enthält dementsprechend die IDs = "30,28,40". Für die Regelauswertung werden die Antworten des App-Anwenders aus der Tabelle „answers“ zu diesen Fragen abgerufen. Anschließend werden die Antworten an die richtigen Stellen in die Regel eingesetzt, z.B. überschreibt die Antwort zu Frage 30 die ID der Frage. Die Regel wird von der eval-Funktion interpretiert. Die eval-Funktion ergibt in diesem Beispiel, dass die Regel wahr ("true") ist und für den App-Anwender zutrifft. Für den Fall, dass beispielsweise beide Vergleiche vor der Und-Verknüpfung für einen App-Anwender nicht zutreffen ('false'=='true') oder der Vergleich nach der Und-Verknüpfung nicht zutrifft ('noise'=='tone'), ergibt die Durchführung der eval-Funktion, dass die Regel und das entsprechende Feedback für einen App-Anwender nicht zutrifft ("false").

Erstellte Regel aus der Datenbank:

`((30 == 'true') || (28 == 'true')) && (40 == 'tone')`

Regel nach Einfügen der Antworten des App-Anwenders:

`(('true' == 'true') || ('true' == 'true')) && ('tone' == 'tone')`

Abbildung 5.17.: Beispiel - Regelauswertung

Trifft die Regel zu, wird das Feedback in dem Array gespeichert, das alle Feedbacks beinhaltet, die für den Tipp der Woche gegeben werden können. Aus diesem Array wird für den Tipp der Woche ein Feedback ermittelt, indem überprüft wird, welche Feedbacks der App-Anwender noch nicht erhalten hat. Zur Visualisierung für die App wird das Feedback an die Schnittstelle zurückgegeben (vgl. Kapitel 5.5.2).

6

Anforderungsabgleich

Dieses Kapitel beschreibt, ob die in der Anforderungsdefinition (vgl. Kapitel 3.4) festgelegten Anforderungen zur Erweiterung der TrackYourTinnitus-Plattform realisiert sind. Kapitel 6.1 stellt den Erfüllungsgrad der funktionalen Anforderungen vor und Kapitel 6.2 den Grad der Erfüllung der nichtfunktionalen Anforderungen.

6.1. Funktionale Anforderungen

Nachfolgend werden die realisierten funktionalen Anforderungen mit den Zielvorgaben aus der Anforderungsdefinition (vgl. Kapitel 3.4.1) verglichen. Tabelle 6.1 listet diese Anforderungen auf, indem für jede Anforderung Identifikationsnummer (ID), Bezeichnung, das geforderte Erfüllungskriterium ("Kr.") und der Erfüllungsstatus ("Status") angegeben sind. Beim Erfüllungsgrad der definierten Anforderungen werden die Stati „erfüllt“, „teilweise erfüllt“ und „nicht erfüllt“ unterschieden.

Die funktionalen Anforderungen FA-01 bis FA-05 sind durch das entwickelte Feedback-Formular (vgl. Kapitel 4.5.1) realisiert. Administratoren sind über das Feedback-Formular in der Lage, alle in Tabelle 4.3 aufgelisteten Feedback-Arten (vgl. Kapitel 4.3) anzulegen. Hierzu zählt das Anlegen von allgemeinem Feedback (FA-01), individuellem Feedback (FA-02), Hauptgruppen-Feedback (FA-03), Subgruppen-Feedback (FA-04) und bei Bedarf das Erstellen entsprechender Regeln (FA-05). Weiter sind die Anforderungen FA-06 und FA-07 durch eine Übersichtsseite der angelegten Feedbacks im Adminbereich umgesetzt. Über diese Seite können alle angelegten Feedbacks in Form einer Liste eingesehen werden (FA-06) und jedes erstellte Feedback kann bei Bedarf gelöscht werden (FA-07).

Innerhalb der Android-Applikation sind die Anforderungen FA-08 bis FA-16 umgesetzt. Ein App-Anwender kann Allgemeines-Feedback (FA-08), Hauptgruppen- (FA-09) und Subgruppen-Feedback (FA-10) erhalten, indem jede Woche eine dieser Feedback-Arten im Tipp der Woche (FA-11) angezeigt wird. Ein App-Anwender kann die Hauptgruppen-Feedbacks für seine Einflussfaktoren sortiert einsehen (FA-12). Je nach Datenberechnung im Hauptgruppen-Algorithmus (vgl. Kapitel 5.6.1) und der daraus resultierender Farbkodierung für die Darstellung der Datenzuverlässigkeit der Einflussfaktoren bedeutet dies, dass jene mit grüner Farbkodierung an oberster Stelle angezeigt werden, anschließend Einflussfaktoren mit gelber und darauffolgend mit grauer Farbkodierung. Zudem kann der App-Anwender zur Berechnung der Feedbacks für die Einflussfaktoren den Datumsbereich ändern (FA-13), sodass für die Berechnung im Hauptgruppen-Algorithmus sein eingegebenes Zeitintervall berücksichtigt wird. Zudem kann der App-Anwender jedes erhaltene Feedback dahingehend bewerten (FA-14), ob es hilfreich oder nicht hilfreich ist. Weiter kann er selbst Feedback geben (FA-15), indem er

Wünsche und Anregungen per E-Mail versenden kann. Feedbacks der Ansicht Einflussfaktoren und Tipp der Woche werden in der App gespeichert (FA-16), damit diese Feedbacks auch dann angezeigt werden können, wenn kein Internet vorhanden ist. Sobald Internet verfügbar ist und neue Feedbacks angefordert werden, werden die neuen Feedbacks gespeichert.

Der Feedback-Algorithmus ermittelt die Hauptgruppen (FA-17), zu denen der App-Anwender auf Basis der Antworten aus dem App-Fragebogen gehört und wertet die vom Administrator erstellten Regeln aus (FA-18), um den Tipp der Woche geben zu können. Für die Einflussfaktoren ermittelt der Feedback-Algorithmus zudem die Datenzuverlässigkeit (FA-19) und markiert in der Datenbank durch setzen von Flags (FA-20), welche Antworten aus dem App-Fragebogen in die Berechnung der Hauptgruppen und die Berechnung der Datenzuverlässigkeit eingeflossen sind. Damit zur Regeldefinition im Feedback-Formular (vgl. Kapitel 4.5.1) und zur Regelauswertung im Feedback-Algorithmus alle Fragetypen der verschiedenen Fragebögen berücksichtigt werden können, müssen auch alle Fragen in der Datenbank angelegt sein. Für den Fragebogen TSCHQ ist die Datenbank um die Fragen 15-35 ergänzt worden (FA-21).

Tabelle 6.1.: Erfüllungsgrad funktionale Anforderungen

ID	Bezeichnung	Beschreibung	Kr.	Status
Adminbereich				
FA-01	Allgemeines-Feedback erstellen	Siehe Kapitel 4.5.1	M	erfüllt
FA-02	Individuelles-Feedback erstellen	Siehe Kapitel 4.5.1	M	erfüllt
FA-03	Hauptgruppen-Feedback erstellen	Siehe Kapitel 4.5.1	M	erfüllt
FA-04	Subgruppen-Feedback erstellen	Siehe Kapitel 4.5.1	M	erfüllt
FA-05	Regeln erstellen	Siehe Kapitel 4.5.1	M	erfüllt
FA-06	Feedbacks einsehen	Siehe Kapitel 4.5.1	K	erfüllt
FA-07	Feedbacks löschen	Siehe Kapitel 4.5.1	K	erfüllt
Android Applikation				
FA-08	Allgemeines-Feedback erhalten	Siehe Kapitel 5.4.2	M	erfüllt
FA-09	Hauptgruppen-Feedback erhalten	Siehe Kapitel 5.4.1	M	erfüllt
FA-10	Subgruppen-Feedback erhalten	Siehe Kapitel 5.4.2	M	erfüllt
FA-11	Tipp der Woche erhalten	Siehe Kapitel 5.4.2	M	erfüllt
FA-12	Einflussfaktoren sortiert einsehen	Siehe Kapitel 5.4.1	M	erfüllt
FA-13	Datumsbereich ändern	Siehe Kapitel 5.6.1	M	erfüllt
FA-14	Feedback bewerten	Siehe Kapitel 5.4.3	M	erfüllt
FA-15	Feedback geben	Siehe Kapitel 5.4.4	K	erfüllt
FA-16	Feedback speichern	Siehe Kapitel 5.4.2	K	erfüllt

Feedback-Algorithmus				
FA-17	Hauptgruppen ermitteln	Siehe Kapitel 5.6.1	M	erfüllt
FA-18	Regeln auswerten	Siehe Kapitel 5.6.2	M	erfüllt
FA-19	Datenzuverlässigkeit berechnen	Siehe Kapitel 5.6.1	M	erfüllt
FA-20	Flags setzen	Siehe Kapitel 5.6.1	K	erfüllt
FA-21	Statistische Fragebögen vollständig bereitstellen	Siehe Kapitel 3.4.1	M	erfüllt

6.2. Nichtfunktionale Anforderungen

Der Anforderungsabgleich für die nichtfunktionalen Anforderungen erfolgt nach demselben Prinzip, wie in Kapitel 6.1. In Tabelle 6.2 sind diese Anforderungen mit Erfüllungsstatus aufgelistet.

Tabelle 6.2.: Erfüllungsgrad nichtfunktionale Anforderungen

ID	Bezeichnung	Beschreibung	Kr.	Status
Datenbereitstellung, -verarbeitung und -speicherung				
NFA-01	Hauptgruppendefinition	Siehe Kapitel 4.2	M	erfüllt
NFA-02	Verarbeitung korrekter App-Datensätze	Siehe Kapitel 5.6.1	M	erfüllt
NFA-03	Internationalisierung	Siehe Kapitel 5.2	M	erfüllt
NFA-04	Datenspeicherung der Hauptgruppen	Siehe Kapitel 5.2	M	erfüllt
NFA-05	Korrekte Datenhaltung	Siehe Kapitel 3.4.2	M	erfüllt
Grafische Benutzeroberfläche				
NFA-06	Listen-basierte Feedback-Darstellung	Siehe Kapitel 4.5.2	M	erfüllt
NFA-07	Detaillierte Feedback-Darstellung	Siehe Kapitel 4.5.2	M	erfüllt
NFA-08	Datenzuverlässigkeit grafisch darstellen	Siehe Kapitel 4.5.2	K	erfüllt
NFA-09	Design	Siehe Kapitel 4.5	M	erfüllt
NFA-10	Mehrsprachigkeit der Feedbacks	Siehe Kapitel 4.5	M	erfüllt

Zur Auswahl von Hauptgruppen im Feedback-Formular und zur Zuweisung eines Benutzers zu einer Hauptgruppe im Hauptgruppen-Algorithmus, sind die Hauptgruppen im Vorfeld definiert (NFA-01) und in der Datenbank angelegt. Unvollständige App-Fragebögen werden im Hauptgruppen-Algorithmus heraus gefiltert, damit nur korrekte App-Fragebögen verarbeitet werden (NFA-02). Alle Feedback-Arten lassen sich im Feedback-Formular für die Sprachen Englisch und Deutsch erstellen (NFA-03). Für spätere Berechnungen, werden die Hauptgruppen, denen ein App-Anwender zugewiesen ist, während der Durchführung des Hauptgruppen-Algorithmus in die Datenbank gespeichert (NFA-04). Zur korrekten Bereitstellung der Fragen der statistischen Fragebögen (NFA-05), ist die Abfrage der

Pflichtfragen bzw. der optionalen Fragen richtig angepasst worden. Dies bedeutet die statistischen Fragebögen werden erst dann in der Datenbank gespeichert, wenn alle Pflichtfragen ausgefüllt und die Fragebögen damit vollständig sind. In der App werden Feedbacks für die Einflussfaktoren listen-basiert (NFA-06) und einzeln ausgewählte Feedbacks detailliert (NFA-07) dargestellt. Die Datenzuverlässigkeit wird durch den berechneten DOR-Wert grafisch dargestellt (NFA-08). Das Design der Android-Applikation und des Feedback-Formulars orientieren sich am Design der bestehenden TrackYourTinnitus-Plattform (NFA-09). Die Benutzeroberfläche wird dem Anwender in der richtigen Sprache angezeigt (NFA-10).

Zusammenfassung und Ausblick

Unter Einbezug der gesetzten Ziele, fasst Kapitel 7.1 die Arbeit zusammen. Kapitel 7.2 gibt zudem einen Ausblick auf zukünftige Erweiterungspotentiale für den Feedback-Algorithmus, die App-Funktionalität sowie für das Projekt TrackYourTinnitus.

7.1. Zusammenfassung

Ziel der Arbeit war es die bestehende Mobile Crowd Sensing Plattform TrackYourTinnitus durch die Entwicklung einer Feedback-Funktionalität für Tinnitus-Betroffene zu erweitern. TrackYourTinnitus hat Tinnitus-Betroffenen bisher erlaubt die Schwankungen der persönlichen Tinnitus-Wahrnehmung zu dokumentieren. Die Erfassung der Schwankungen der Tinnitus-Wahrnehmung erfolgt über Apps für die Betriebssysteme Android und iOS. Zudem existiert eine Webseite, über die die Anmeldung für TrackYourTinnitus vorgenommen werden kann und über die zu Beginn drei statistische Fragebögen auszufüllen sind. Die Anzahl an etwa drei Millionen Tinnitus-Betroffenen allein in Deutschland bestätigt die Relevanz der Tinnitus-Forschung [Tin15]. Aufgrund dessen beschäftigt sich diese Arbeit mit der Frage, wie eine mobile Applikation eingesetzt werden kann, um Tinnitus-Betroffenen Rückmeldung darüber geben zu können, was ihnen im Umgang mit Tinnitus bzw. zur Linderung des Tinnitus helfen kann. Diese Arbeit hat sich mit der Forschungsfrage: "Wie muss ein Feedback-Algorithmus für Tinnitus-Betroffene implementiert sein, um über eine mobile Applikation Feedbacks zu geben?" befasst. Als Ergebnis ist ein Feedback-Algorithmus entstanden.

Neben den relevanten Grundlagen für die Arbeit, wie beispielsweise die Einführung in das Themengebiet Tinnitus und Mobile Crowd Sensing, ist innerhalb dieser Arbeit ein Konzept entwickelt worden, das der Erweiterung von TrackYourTinnitus um einen Feedback-Algorithmus dient. Zur Beantwortung der Forschungsfrage sind mobile Applikationen aus dem Gesundheitsbereich auf Feedback-Funktionalitäten sowie der Ist-Zustand der Daten der TrackYourTinnitus-Plattform analysiert worden. Weiter diente zur Funktionsanalyse die Methodik der Anwendungsfälle, wodurch Anforderungen aus Benutzersicht betrachtet werden. Zur Verfeinerung der Ziele diente ein Szenario für den geplanten Ablauf des Feedback-Algorithmus. Aufbauend sind die funktionalen und nichtfunktionalen Anforderungen erfasst und über die Projektlaufzeit aktualisiert worden.

Das erarbeitete Konzept beschreibt vier verschiedene Umsetzungsoptionen, die sich zur Realisierung eines Feedback-Algorithmus eignen und beantwortet die Forschungsfrage, wie ein Feedback-Algorithmus implementiert sein muss. Zudem berücksichtigt das entwickelte Konzept, wie Tinnitus-Betroffene gruppiert werden können, um Feedbacks für bestimmte Tinnitus-Gruppen geben zu können. Darüber hinaus erarbeitet das Konzept vier verschiedene

Feedback-Arten, die Tinnitus-Betroffenen gegeben werden können. Ferner ist im Konzept berücksichtigt worden, wie Administratoren diese Feedback-Arten sinnvoll anlegen können. Einen wichtigen Bestandteil stellt das entwickelte Konzept des Feedback-Algorithmus dar. In diesem Konzept führen alle erarbeiteten Konzepte zusammen, denn der Feedback-Algorithmus muss einzelne App-Anwender zu Tinnitus-Gruppen zuweisen. Weiter muss er für diese Gruppen überprüfen, welche Feedbacks gegeben werden können und gegebenenfalls hierzu Regeln auswerten. Aus diesem Grund ist für Administratoren zudem ein Konzept in Form eines Feedback-Formulars für die Regelerstellung zur Regelauswertung erarbeitet worden sowie für App-Anwender ein Navigationskonzept für die Visualisierung von Feedbacks innerhalb der Android-Applikation.

Die erarbeiteten Konzepte wurden innerhalb dieser Arbeit umgesetzt. Dazu wurde zunächst die Datenbank der TYT-Plattform um die erforderlichen Tabellen für die Tinnitus-Gruppen und Feedbacks erweitert. Das konzipierte Feedback-Formular ist im Adminbereich der Webseite umgesetzt und die bestehende Android-Applikation um entsprechende Aktivitäten erweitert, die zur Einsicht von Feedbacks in der Einflussfaktoren-Ansicht dienen. Zur Kommunikation der App mit der TYT-Plattform sind neue REST-Schnittstellen entwickelt worden, über die der Feedback-Algorithmus ausgeführt wird. Der Feedback-Algorithmus sowie die Regelauswertung wurden als neue Komponenten der TYT-Plattform implementiert. Hauptpunkt der Arbeit ist der entwickelte Feedback-Algorithmus, der alle aus dem Konzept geforderten Anforderungen erfüllt. Er verarbeitet die Daten für jeden App-Anwender einzeln und weist ihn zu Hauptgruppen zu. Für diese Hauptgruppen werden Feedbacks gegeben. Diese Feedbacks werden in der App als Tipp der Woche angezeigt. Bei Bedarf findet eine Regelauswertung statt, die zusätzlich prüft, ob Feedbacks einem App-Anwender gegeben werden können. Zudem stellt diese Arbeit einen Anforderungsvergleich auf, der zeigt, dass alle geforderten Anforderungen umgesetzt sind.

Diese Arbeit beantwortet die Forschungsfrage durch das erarbeitete Konzept für die TYT-Plattform, indem die App-Fragebögen dazu verwendet werden Tinnitus-Betroffene Gruppen zuzuweisen. Anhand dieser Gruppenzuweisung und weiterer Regelauswertungen zu den statistischen Fragebögen kann Tinnitus-Betroffenen über eine mobile Applikation Feedback basierend auf ihren ausgefüllten Fragebögen gegeben werden.

7.2. Ausblick

Nachfolgend werden verschiedene Erweiterungspotentiale für den Feedback-Algorithmus, die App-Funktionalität sowie für die TrackYourTinnitus-Plattform beschrieben. Dabei wird erläutert, wie diese in zukünftigen Arbeiten zum Einsatz kommen können.

7.2.1. Erweiterung des Feedback-Algorithmus

Zusätzlich zum realisierten Konzept gibt es ein großes Erweiterungspotential für den Feedback-Algorithmus. Das in Kapitel 4.1 erarbeitete Konzept lässt sich beispielsweise dadurch erweitern, dass jeder Einflussfaktor durch den Feedback-Algorithmus weiter analysiert wird, indem ermittelt wird, wie viele Faktoren den Einflussfaktor bestimmen und in welchem Ausmaß. Beispielsweise lässt sich für den Einflussfaktor Stress weiterhin ermitteln, welche

Stressfaktoren ihn charakterisieren und zu dem Leiden des Tinnitus-Betroffenen beitragen. Umsetzen lässt sich dies beispielsweise dadurch, dass weitere Fragebögen zu den bestehenden statistischen Fragebögen hinzugezogen werden. Nach demselben Prinzip lässt sich dies auf die restlichen Einflussfaktoren übertragen. Auf Basis der gewonnenen Erkenntnisse, lassen sich neue Feedbacks für Tinnitus-Betroffene geben.

Im Weiteren kann der Feedback-Algorithmus dahingehend erweitert werden, dass er die hilfreichsten Feedbacks für die verschiedenen Tinnitus-Gruppen ermittelt und diese zuerst in der App angezeigt werden. Hierzu lässt sich für die bewerteten Feedbacks der App-Anwender eine Rangliste erstellen. Jedes als hilfreich bewertete Feedback steigt innerhalb der Rangliste einen Rang auf, während Feedbacks, die als nicht hilfreich bewertet werden einen Rang nach unten verschoben werden. Dadurch kann ermittelt werden, welches Feedback insgesamt am hilfreichsten ist. Zum anderen kann auch ermittelt werden, welche Feedbacks bezogen auf die definierten Tinnitus-Gruppen am meisten helfen. Daher können beispielsweise für jede Tinnitus-Gruppe jeweils die besten drei ermittelten Feedbacks aus der Rangliste gegeben werden. Kommt es vor, dass für einen Tinnitus-Betroffenen alle drei Feedbacks je Tinnitus-Gruppe nicht hilfreich sind, können die nächsten drei Feedbacks aus der Rangliste gegeben werden.

Ferner lässt sich der Feedback-Algorithmus dahingehend erweitern, dass für die Tinnitus-Gruppen auch Therapie-Feedbacks gegeben werden. Durch weitere Fragebögen zu bisher durchgeführten Therapien, lassen sich Regeln zu den Therapie-Feedbacks erstellen. Beispielsweise kann abgefragt werden, welche Therapien bisher durchgeführt worden sind und welche nicht erfolgreich waren. Durch eine entsprechende Regelauswertung können anschließend weitere Therapie-Vorschläge unter Berücksichtigung der nicht hilfreichen Therapien gegeben werden. Für jede Tinnitus-Gruppe kann zudem eine Therapie-Rangliste erstellt werden. Dadurch können zukünftig Rückschlüsse dahingehend gezogen werden, welche Therapie für welche Tinnitus-Gruppe auf Grundlage der Therapie-Rangliste am besten geeignet ist.

7.2.2. Erweiterung der App-Funktionalität

App-Anwender können zukünftig die erhaltenen Feedbacks in einer Favoriten-Liste speichern, um Feedbacks aus dem Tipp der Woche nach Ablauf einer Woche zu einem späteren Zeitpunkt erneut einsehen zu können. Weiter können die Feedbacks anderen Personen per E-Mail, über soziale Netzwerke oder innerhalb einer TrackYourTinnitus-Community empfohlen werden.

Um einen Anreiz dafür zu schaffen, dass App-Anwender die App regelmäßig verwenden, kann die App um eine Tagebuch-Funktion in Form eines Kalenders erweitert werden. Innerhalb des Kalenders können Kalendereinträge beispielsweise zur Dokumentation von Therapie-Verläufen eingesetzt werden, sodass der App-Anwender einen laufenden Überblick über bereits unternommene Therapien hat. Dies erleichtert das Zurückverfolgen über Therapiedauer und Therapieverlauf. Beim Anlegen einer neuen Therapie, kann der entwickelte Feedback-Algorithmus dazu verwendet werden, um zu ermitteln, welchen Tinnitus-Gruppen ein App-Anwender vor Therapiebeginn zugewiesen ist. Jeder ausgefüllte App-Fragebogen und die dadurch dokumentierten Schwankungen der Tinnitus-Wahrnehmung lassen sich dem Therapie-Eintrag zuordnen. Dadurch werden die Schwankungen der Tinnitus-Wahrnehmung nicht mehr einzeln betrachtet, sondern in Bezug zu einer Therapie gestellt. Zusätzlich

können immer beim Anlegen eines neuen Therapie-Eintrags sowie bei Beendigung einer Therapie weitere Fragebögen verwendet werden, um Rückschlüsse auf den Zustand, die Lebensphase und Therapie-Erfolg des Tinnitus-Betroffenen zu erhalten. Alle Feedbacks, die der App-Anwender innerhalb einer Therapie als hilfreich bewertet, können zudem auch automatisch der Therapie zugeordnet werden, sodass am Ende der Therapie ein entsprechender Bericht gegeben werden kann. Das Ende einer Therapien sollte durch einen entsprechenden Kalendereintrag gekennzeichnet sein, damit der Zeitraum für die Dauer der Therapie bekannt ist und dieses Zeitintervall beispielsweise für den Hauptgruppen-Algorithmus verfügbar ist, um zu ermitteln, welchen Hauptgruppen ein Tinnitus-Betroffener während einer Therapie zugeordnet ist. Innerhalb des Berichts können die zu Beginn ermittelten Einflussfaktoren genannt werden, sowie die Gruppen, zu denen der App-Anwender während der Therapie gehört. Weiter kann der Algorithmus auch so definiert werden, dass er ermittelt zu welchen Gruppen der App-Anwender nach einer definierten Zeitspanne nach der Ende der Therapie gehört. Im Bericht können alle ermittelten hilfreichen Feedbacks gelistet sein. Zudem kann der App-Anwender die durchgeführten Therapien bewerten, um so ermitteln zu können, welche Therapie-Form bei welcher Tinnitus-Gruppe am hilfreichsten ist oder die Therapie weiter empfehlen.

Zudem kann auch über weitere Fragebögen ermittelt werden, was die Tinnitus-Betroffenen selbst vermuten, was ihre persönlichen Einflussfaktoren sind und welche Maßnahmen im Umgang mit dem Tinnitus helfen.

7.2.3. Erweiterung der TrackYourTinnitus-Plattform

Die Feedback-Funktionalität, die für Tinnitus-Betroffene bisher für Android realisiert ist, lässt sich auch auf iOS und Windows übertragen, da der entwickelte Feedback-Algorithmus nicht auf der Android-App entwickelt wurde, sondern auf der TYT-Plattform.

Das Projekt TrackYourTinnitus lässt sich durch eine Online-Community zudem dahingehend erweitern, dass alle am System angemeldeten Tinnitus-Betroffene beitreten können. App-Anwender, die über die Feedback-Funktionalität verfügen und Tinnitus-Gruppen zugewiesen sind, sind innerhalb der Community ihren Tinnitus-Gruppen zugewiesen und können sich mit anderen Betroffenen austauschen.

A

Anhang

Tabelle A.1.: App-Fragebogen [Sch14]

Nr.	Frage
01	Haben Sie gerade den Tinnitus bewusst wahrgenommen?
02	Wie laut ist der Tinnitus momentan?
03	Wie belastend finden Sie den Tinnitus momentan?
04	Wie ist Ihre aktuelle Stimmungslage?
05	Wie aufgeregt sind Sie gerade?
06	Wie gestresst fühlen Sie sich gerade?
07	Wie sehr haben Sie sich auch das konzentriert, was Sie gerade tun?

Tabelle A.2.: Mini Tinnitus Fragebogen (Mini-TF) [HG04]

Nr.	Frage
01	Ich bin mir der Ohrgeräusche vom Aufwachen bis zum Schlafengehen bewußt.
02	Ich mache mir wegen der Ohrgeräusche Sorgen, ob mit meinem Körper ernstlich etwas nicht in Ordnung ist.
03	Wenn die Ohrgeräusche andauern, wird mein Leben nicht mehr lebenswert sein.
04	Auf Grund der Ohrgeräusche bin ich mit meiner Familie und meinen Freunden gereizter.
05	Ich Sorge mich, dass die Ohrgeräusche meine körperliche Gesundheit schädigen könnten.
06	Wegen der Ohrgeräusche fällt es mir schwer, mich zu entspannen.
07	Oft sind meine Ohrgeräusche so schlimm, dass ich sie nicht ignorieren kann.
08	Wegen der Ohrgeräusche brauche ich länger zum Einschlafen.
09	Wegen der Ohrgeräusche bin ich leichter niedergeschlagen.
10	Ich denke oft darüber nach, ob die Ohrgeräusche jemals weggehen werden.
11	Ich bin Opfer meiner Ohrgeräusche.
12	Die Ohrgeräusche haben meine Konzentration beeinträchtigt.

Tabelle A.3.: Tinnitus Sample Case History Questionnaire (TSCHQ) [Lan06]

Nr.	Frage	Antwortmöglichkeiten
01	Geburtsdatum	Datum
02	Geschlecht	männlich; weiblich
03	Händigkeit	rechts; links; beidhändig
04	Tinnitus-Beschwerden in der Familie	JA; NEIN; Wenn JA: Eltern; Geschwister; Kinder
05	Beginn des Tinnitus: Wann haben Sie den Tinnitus zum ersten Mal wahrgenommen?	Text
06	Wie haben Sie den Beginn wahrgenommen?	allmählich; unvermittelt
07	Stand der Beginn Ihres Tinnitus in Verbindung mit:	Verletzung der Halswirbelsäule; Knalltrauma; Veränderung des Hörvermögens; Stress; Kopfverletzung; Sonstiges
08	Haben Sie das Gefühl, dass Ihr Tinnitus pulsiert?	JA, im Rhythmus meines Herzschlags; JA, anders als mein Herzschlag; NEIN
09	Wo nehmen Sie Ihren Tinnitus wahr?	im rechten Ohr; im linken Ohr; in beiden Ohren, stärker im linken Ohr; in beiden Ohren, stärker im rechten Ohr; in beiden Ohren gleich stark, im Inneren des Kopfes; an anderer Stelle
10	Wie ist der Verlauf?	Phasen mit und ohne Tinnitus wechseln sich ab; Der Tinnitus ist ständig vorhanden
11	Verändert sich die LAUTSTÄRKE Ihres Tinnitus von Tag zu Tag?	JA; NEIN
12	Bitte beschreiben Sie die durchschnittliche LAUTSTÄRKE Ihres Tinnitus.	Skala von 1-100
13	Bitte beschreiben Sie mit eigenen Worten, wie Ihr Tinnitus normaler Weise klingt:	Text
14	Hört sich Ihr Tinnitus eher wie ein Ton an oder eher wie Lärm?	Ton; Lärm; Grillen; andere Empfindung
15	Bitte beschreiben Sie die Frequenz Ihres Tinnitus.	sehr hohe Frequenz; hohe Frequenz; mittlere Frequenz; niedrige Frequenz
16	Wieviele Prozent der Zeit waren Sie sich im letzten Monat Ihres Tinnitus bewußt?	Skala in %
17	Wieviele Prozent der Zeit im letzten Monat haben Sie sich über Ihren Tinnitus geärgert, bzw. waren Sie wegen des Tinnitus unglücklich oder genervt?	Skala in %

18	Wie vielen verschiedenen Behandlungen haben Sie sich unterzogen aufgrund Ihres Tinnitus?	keiner; einer; mehreren; vielen
19	Wird die Lautstärke Ihres Tinnitus durch bestimmte Arten von Umgebungsgeräuschen reduziert bzw. überdeckt, wie zum Beispiel durch das Rauschen eines Wasserfalls oder das Geräusch fließenden Wassers, wenn Sie unter der Dusche stehen?	JA; NEIN; Ich weiß es nicht.
20	Verschlechtert starker Lärm Ihren Tinnitus?	JA; NEIN; Ich weiß es nicht.
21	Beeinflusst eine Bewegung Ihres Kopfes und/oder Ihres Nackens (zum Beispiel das Verschieben des Kiefers oder das Zusammenbeißen der Zähne) oder eine Berührung Ihrer Arme, Hände oder Ihres Kopfes Ihren Tinnitus?	JA; NEIN
22	Beeinflusst ein kurzer Schlaf während des Tages (z.B. Mittagsschlaf) Ihren Tinnitus?	verstärkt meinen Tinnitus; vermindert meinen Tinnitus; hat keine Auswirkung
23	Besteht eine Verbindung zwischen Ihrem Nachtschlaf und Ihrem Tinnitus während des Tages?	JA; NEIN; Ich weiß es nicht.
24	Hat Stress Einfluss auf Ihren Tinnitus?	verstärkt meinen Tinnitus; vermindert meinen Tinnitus; hat keine Auswirkung
25	Beeinflusst die Behandlung mit Medikamenten Ihren Tinnitus?	Text
26	Haben Sie ein Problem mit Ihrem Hörvermögen?	JA; NEIN
27	Benutzen Sie Hörgeräte?	rechts; links; auf beiden Seiten; auf keiner Seite
28	Fühlen Sie sich besonders geräuschempfindlich? Fühlen Sie sich beispielsweise gestört durch Geräusche, die anderen Menschen in Ihrer Umgebung nicht störend laut vorkommen?	niemals; selten; manchmal; gewöhnlich; immer
29	Führen laute Geräusche bei Ihnen zu Schmerz-ähnlichem Empfinden oder zu körperlichem Unwohlsein?	JA; NEIN; Ich weiß es nicht.
30	Leiden Sie unter Kopfschmerzen?	JA; NEIN
31	Leiden Sie unter Schwindel?	JA; NEIN
32	Haben Sie Beschwerden im Bereich Ihres Kiefergelenkes oder Ihrer Kaumuskulatur?	JA; NEIN
33	Leiden Sie unter Nackenschmerzen?	JA; NEIN
34	Leiden Sie unter anderen Schmerzen?	JA; NEIN
35	Befinden Sie Sich momentan in psychiatrischer Behandlung?	JA; NEIN

Tabelle A.4.: Schlimmstes Symptom [Sch13]

Nr.	Symptom
01	Aufgrund des Tinnitus bin ich mit meiner Familie, Freunde und Arbeitskollegen gereizter.
02	Wegen des Tinnitus fällt es mir schwer mich zu entspannen.
03	Ich mache mir starke Sorgen wegen meines Tinnitus.
04	Wegen des Tinnitus fällt es mir schwer abends einzuschlafen.
05	Aufgrund des Tinnitus kann ich mich schlechter konzentrieren.
06	Wegen des Tinnitus bin ich geräuschempfindlicher als früher.
07	Mir fällt es wegen des Tinnitus schwer Gesprächen, Musik oder Filmen zuzuhören.
08	Wegen des Tinnitus fühle ich mich niedergeschlagen.
09	Ich habe keines dieser Symptome.

Literaturverzeichnis

- [Aki14] AKILI:INNOVATION: *metappolic*. Website, 2014. – <https://www.microsoft.com/en-us/store/apps/metappolic/9nblgggzk8f7>; Abgerufen am 18.12.2015.
- [Bac12] BACH, M.: *Mobile Anwendungen mit Android. Entwicklung und praktischer Einsatz*. München : Addison-Wesley, 2012
- [Bie05] BIESINGER, E. AND IRO, H.: *HNO Praxis heute: Tinnitus*. Heidelberg : Springer, 2005
- [Bie07] BIESINGER, E.: *Tinnitus - endlich Ruhe im Ohr. Ursachen erkennen und ausschalten*. TRIAS, 2007 (TRIAS-Therapie-Kompass)
- [BKI06] BEIERLE, C. ; KERN-ISBERNER, G.: *Methoden wissensbasierter Systeme - Grundlagen, Algorithmen, Anwendungen*. Heidelberg : Vieweg-Verlag, 2006
- [CL14] CLEVE, J. ; LÄMMEL, U.: *Data Mining*. Oldenbourg : De Gruyter, 2014 (De Gruyter Studium)
- [Des14a] DESTATIS: *63 % der Internetnutzer/-innen surfen auch mobil*. Website, 2014. – https://www.destatis.de/DE/PresseService/Presse/Pressemitteilungen/2014/12/PD14_457_63931; Abgerufen am 28.11.2015.
- [Des14b] DESTATIS: *80 % (58,6 Millionen) der Personen ab zehn Jahren nutzten im ersten Quartal 2014 das Internet*. Website, 2014. – https://www.destatis.de/DE/ZahlenFakten/GesellschaftStaat/EinkommenKonsumLebensbedingungen/ITNutzung/Aktuell_ITNutzung.html; Abgerufen am 28.11.2015.
- [Eur15] EUROPÄISCHE KOMMISSION: *Die Vorteile von Mobilgeräte-Applikationen für Ihre Gesundheit - 10/04/2014*. Website, 2015. – http://ec.europa.eu/news/environment/140410_de.htm; Abgerufen am 28.11.2015.
- [Goo15] GOOGLE INC: *ExpandableListAdapter Class Referenz*. Website, 2015. – <http://developer.android.com/reference/android/widget/ExpandableListAdapter.html>; Abgerufen am 14.12.2015.
- [Hü06] HÜLSER, J. AND ZIMMERMANN, H.: *Statistische Prinzipien für medizinische Projekte*. 4. Bern : Huber, 2006
- [HG04] HILLER, W. ; GOEBEL, G: *Mini Tinnitus Fragebogen (Mini TF12)*. Website, 2004. – http://www.eutinnitus.com/Dateien%20MiniTF12/MiTiTe_de.php; Abgerufen am 14.12.2015.
- [Inf14] INFERMEDICA: *Symptomate Symptom Checker*. Website, 2014. – <https://play.google.com/store/apps/details?id=com.symptomate.mobile>; Abgerufen am 18.12.2015.
- [IRLP+13] ISELE, D. ; RUF-LEUSCHNER, M. ; PRYSS, R. ; SCHAUER, M. ; REICHERT, M. ; SCHOBEL, J. ; SCHINDLER, A. ; ELBERT, T.: Detecting adverse childhood experiences with a little help from tablet computers. In: *XIII Congress of European Society of Traumatic Stress Studies (ESTSS) Conference*, 2013, S. 69–70

- [KVL13] KREUZER, P. M. ; VIELSMEIER, V. ; LANGGUTH, B.: Chronischer Tinnitus – eine interdisziplinäre Herausforderung. In: *Deutsches Ärzteblatt* 110 (2013), Nr. 16, S. 278–284
- [Lan06] LANGGUTH, B.: *Tinnitus Sample Case History Questionnaire*. PDF, 2006. – http://www.tinnitusresearch.org/en/consensus/consensusdocuments/de/Case_History_Questionnaire_de.pdf; Abgerufen am 14.12.2015.
- [LML⁺10] LANE, N.D. ; MILUZZO, E. ; LU, H. ; PEEBLES, D. ; CHOUDHURY, T. ; CAMPBELL, A.T.: A survey of mobile phone sensing. In: *Communications Magazine, IEEE* 48 (2010), Nr. 9, S. 140–150
- [Mat05] MATTERN F.: Leben und Lernen in einer von Informationstechnologie durchdrungenen Welt – Visionen und Erwartungen. In: *Lernplattformen (Web-Based Training 2005)*. EMPA-Akademie, 2005, S. 39–61
- [Mel08] MELZER, I.: *Service-orientierte Architekturen mit Web Services. Konzepte - Standards - Praxis*. Heidelberg : Springer Spektrum, 2008
- [Nie10] NIELSEN, J.: *Usability engineering*. Amsterdam; Heidelberg [u.a.] : Kaufmann, 2010
- [Otw15] OTWELL, T.: *Laravel Documentation*. Website, 2015. – <http://laravel.com/docs/5.1>; Abgerufen am 14.12.2015.
- [Par10] PARTSCH, H.: *Requirements-Engineering systematisch. Modellbildung für softwaregestützte Systeme*. Berlin, Heidelberg : Springer-Verlag, 2010 (eXamen.press)
- [Pf15] PFIZER INC.: *Mein Kopfschmerz*. Website, 2015. – <https://play.google.com/store/apps/details?id=com.pfizer.at.MeinKopfschmerz>; Abgerufen am 18.12.2015.
- [PHP15] PHP: *Documentation*. Website, 2015. – <https://www.php.net/>; Abgerufen am 14.12.2015.
- [PMLR15] PRYSS, R. ; MUNDBROD, N. ; LANGER, D. ; REICHERT, M.: Supporting medical ward rounds through mobile task and process management. In: *Information Systems and e-Business Management* 13 (2015), February, Nr. 1, S. 107–146
- [Poh07] POHL, K.: *Requirements Engineering. Grundlagen, Prinzipien, Techniken*. Heidelberg : dpunkt.verlag, 2007
- [PRH⁺15] PRYSS, R. ; REICHERT, M. ; HERRMANN, J. ; LANGGUTH, B. ; SCHLEE, W.: Mobile Crowd Sensing in Clinical and Psychological Trials ? A Case Study. In: *28th IEEE Int'l Symposium on Computer-Based Medical Systems*, IEEE Computer Society Press, June 2015, S. 23–24
- [PRLS15] PRYSS, R. ; REICHERT, M. ; LANGGUTH, B. ; SCHLEE, W.: Mobile Crowd Sensing Services for Tinnitus Assessment, Therapy and Research. In: *IEEE 4th International Conference on Mobile Services (MS 2015)*, IEEE Computer Society Press, June 2015, S. 352–359
- [Res14] RESEARCH2GUIDANCE: *Umfrage zum erwarteten wirtschaftlichen Potential von mHealth-Apps nach Anwendungsschwerpunkten im Jahr 2014*

- (Häufigkeitsverteilung). Website, 2014. – <http://de.statista.com/statistik/daten/studie/440446/umfrage/wirtschaftliches-potential-von-mhealth-apps-nach-anwendungsschwerpunkten/>; Abgerufen am 28.11.2015.
- [RLPL⁺13] RUF-LEUSCHNER, M. ; PRYSS, R. ; LIEBRECHT, M. ; SCHOBEL, J. ; SPYRIDOU, A. ; REICHERT, M. ; SCHAUER, M.: Preventing further trauma: KINDEX mum screen - assessing and reacting towards psychosocial risk factors in pregnant women with the help of smartphone technologies. In: *XIII Congress of European Society of Traumatic Stress Studies (ESTSS) Conference*, 2013, S. 70–70
- [San15] SANOVATION AG: *Schmerztagebuch - CatchMyPain*. Website, 2015. – <https://play.google.com/store/apps/details?id=com.sanovation.catchmypain.pro>; Abgerufen am 18.12.2015.
- [Sch13] SCHLEE, W.: *TrackYourTinnitus*. Website, 2013. – <https://www.trackyourtinnitus.org/de>; Abgerufen am 14.12.2015.
- [Sch14] SCHLEE, W.: *TrackYourTinnitus App*. Website, 2014. – <https://play.google.com/store/apps/details?id=com.jochenherrmann.trackyourtinnitus>; Abgerufen am 18.12.2015.
- [SHP⁺14a] SCHLEE, W. ; HERRMANN, J. ; PRYSS, R. ; REICHERT, M. ; LANGGUTH, B.: How dynamic is the continuous tinnitus percept? In: *11th International Tinnitus Seminar*, 2014
- [SHP⁺14b] SCHLEE, W. ; HERRMANN, J. ; PRYSS, R. ; REICHERT, M. ; LANGGUTH, B.: Moment-to-moment variability of the auditory phantom perception in chronic tinnitus. In: *13th Int'l Conf on Cochlear Implants and Other Implantable Auditory Technologies*, 2014
- [Son15] SONORMED GMBH: *Tinnitracks Tinnitus Therapie*. Website, 2015. – <https://play.google.com/store/apps/details?id=de.sonormed.tinnitracks&hl=de>; Abgerufen am 18.12.2015.
- [SPR15] SCHOBEL, J. ; PRYSS, R. ; REICHERT, M.: Using Smart Mobile Devices for Collecting Structured Data in Clinical Trials: Results From a Large-Scale Case Study. In: *28th IEEE International Symposium on Computer-Based Medical Systems (CBMS 2015)*, IEEE Computer Society Press, June 2015, S. 13–18
- [SRP⁺15] SCHICKLER, M. ; REICHERT, M. ; PRYSS, R. ; SCHOBEL, J. ; SCHLEE, W. ; LANGGUTH, B.: *Entwicklung mobiler Apps: Konzepte, Anwendungsbausteine und Werkzeuge im Business und E-Health*. Springer Vieweg, 2015 (eXamen.press)
- [SSP⁺14] SCHOBEL, J. ; SCHICKLER, M. ; PRYSS, R. ; MAIER, F. ; REICHERT, M.: Towards Process-Driven Mobile Data Collection Applications: Requirements, Challenges, Lessons Learned. In: *10th Int'l Conference on Web Information Systems and Technologies (WEBIST 2014), Special Session on Business Apps*, 2014, S. 371–382
- [Sta13] STATISTA: *Große Vielfalt im Bereich der Gesundheitsapps*. Website, 2013. – <http://de.statista.com/infografik/2988/anzahl-der-gesundheitsapps-nach-therapiebereich/>; Abgerufen am 28.11.2015.

- [Sta15] STATISTA: *Marktanteile der mobilen Betriebssysteme am Absatz von Smartphones in Deutschland von Januar 2012 bis September 2015*. Website, 2015. – <http://de.statista.com/statistik/daten/studie/225381/umfrage/marktanteile-der-betriebssysteme-am-smartphone-absatz-in-deutschland-zeitreihe/>; Abgerufen am 21.09.2015.
- [TCTJ08] TYLER, R. ; COELHO, C. ; TAO, P. ; JI, H.: Identifying Tinnitus Subgroups With Cluster Analysis. In: *American Journal of Audiology* 17 (2008), Nr. 1, S. 176—184
- [Tin15] TINNITUSZENTRUM UNIVERSITÄT REGENSBURG: *Häufige Fragen zu Tinnitus*. PDF, 2015. – http://www.tinnituszentrum-regensburg.de/htmls/information/Patienteninformation_Tinnitus.pdf; Abgerufen am 28.11.2015.
- [TOM14] TOMORROW FOCUS AG: *Mobile Effects 2014 - 1. Das Leben in der digitalen Welt*. PDF, 2014. – http://www.forward-adgroup.de/uploads/tx_mjstudien/ForwardAdGroup_Studie_MobileEffects_2014-1.pdf; Abgerufen am 28.11.2015.
- [TOM15] TOMORROW FOCUS AG: *Mobile Effects 2015. Always On - Wie das Mobile Web den Alltag verändert*. PDF, 2015. – http://www.forward-adgroup.de/uploads/tx_mjstudien/ForwardAdGroup_Studie_MobileEffects_2015.pdf; Abgerufen am 28.11.2015.
- [Wha12] WHARTON, J.: *ActionBarSherlock*. Website, 2012. – <http://actionbarsherlock.com/>; Abgerufen am 14.12.2015.
- [WIM12] WILLNECKER, F. ; ISMAILOVIĆ, D. ; MAISON, W.: Architekturen mobiler Multiplattform-Apps. In: VERCLAS, S. (Hrsg.) ; LINNHOF-POPIEN, C. (Hrsg.): *Smart Mobile Apps*. Springer Berlin Heidelberg, 2012 (Xpert.press), S. 403–417
- [ZRJT10] ZHANG, X. ; RAS, Z.W. ; JASTREBOFF, P.J. ; THOMPSON, P.L.: From Tinnitus Data to Action Rules and Tinnitus Treatment. In: *Granular Computing (GrC), 2010 IEEE International Conference on*, 2010, S. 620–625